```
                .REPT   0


                                IDENTIFICATION
                                --------------

PRODUCT CODE:   AC-5234B-MC

PRODUCT NAME:   CXVSC80 VSV11 CSS DEC/X MOD

DATE CREATED:   OCT 1,1980

MAINTAINER:     CSS PRODUCTS GROUP

AUTHOR(S):      GUS PASQUANTONIO
                PAUL TAYLOR
                DICK GORDON

REVISION HISTORY:

        PAUL TAYLOR     18-AUG-81       <VERSION A1>
        DICK GORDON     20-DEC-81       <<VER B>>
```



```
ABSTRACT:
--------
        "XVSCB" IS A DEC/X11 SOFTWARE MODULE WHICH EXERCISES A
        VSV11/VS11 GRAPHIC SYSTEM.
        THE "XVSCB" DEC/X11 MODULE CONSISTS OF

        *** VSV-11 INSTRUCTION SET.
        *** SPECIAL MACROS AND OPDEFS.
        *** DEVICE REGISTER ASSIGNMENTS
        *** JOY-STICK AND TIME-OUT INTERRUPT SERVICE ROUTINES.
        *** LOOK-UP-TABLE SERVICE ROUTINES.
        *** PROGRAM START RESTART
        *** PART 1 -- THE STANDARD VISUAL REFERENCE FRAME
        *** PART 2 -- THE OLD BOUNCING BALLS
        ***     SR1 BIT0 = 1 = ENABLE PART  2.
        ***     SR1 BIT1 = 1 = USE TWO MEMORY DYNAMIC MODE.
        ***     SR1 BIT2 = 1 = DISABLE BELL ON BOUNCE.
        ***     SR2 BIT0 = 1 = ENABLE COLOR LOOKUP TABLE        <VERSION A1>
        ***     NOTE:

                THE EXERCISER WILL WORK WITH A SINGLE MEMORY CHANNEL,
        EXCEPT FOR THE COMBINATION OF SR1 BIT0=1 AND SR1 BIT1=1.THE
        VSV11/VS11 CAN HAVE UP TO FOUR SEPARATE MEMORY CHANNELS (0 THRO 3)
        THOSE ARE JUMPER SELECTABLE ON THE MODULE.TO RUN THE TEST WITH
        BOTH SR1 BIT0=1 AND SR1 BIT1=1,TWO MEMORY CHANNELS ARE NEEDED.
        REFER THE VSV TECHNICAL MANUAL (FIGURE 2-4) FOR DYNAMIC
        CONFIGURATION.

                ALSO IF THE COLOR LOOKUP IS SPECIFIED ONLY ONE VSV IS
        EXERCISED. THIS IS DONE BY SETTING BIT 0 OF SR2.
                SR1 = 0         ;TEST PATTERN ONLY
                SR1 = 1         ;DISPLAY+BOUNCING BALL+BELL
                SR1 = 2         ;TEST PATTERN ONLY
                SR1 = 3         ;TEST PATTERN+BOUNCING BALL+BELL(TWO CHANNELS)
                SR1 = 4         ;TEST PATTERN ONLY
                SR1 = 5         ;TEST PATTERN+BOUNCING BALL
                SR1 = 6         ;TEST PATTERN ONLY
                SR1 = 7         ;TEST PATTERN+BOUNCING BALL (TWO CHANNELS)
***

        *** PART 1 -- DISPLAY CODE
        *** PART 2 -- DISPLAY CODE
        *** ASCII CHARACTER ADDRESS TABLE AND SUBPIX
        *** 256 LEVEL LUMINANCE (GREY-SCALE) TABLE
        *** PATCH AREA AND END
```

```
                                REQUIREMENTS:
                                -------------
                                        PDP-11/LSI-11 PROCESSOR WITH 16K MEMORY OR MORE MEMORY
                                        CONSOLE DEVICE (LA30,LA36,VT50,VT100,ETC.)
                                        XXDP + LOAD DEVICE (RX,RP,RL,TM,DT,ETC.)
                                        1 TO 8 VSV11 SYSTEMS, EACH CONSISTS OF:

                                                M7064  IMAGE PROCESSOR
                                                M7062  IMAGE MEMORY (1 TO 4 CHANNELS)
                                                M7069  COLOR LOOK-UP RAM (OPTIONAL)
                                                VCG04  SYNC GENERATOR WITH CURSOR CONTROL
                                                DISPLAY MONITOR, COLOR OR MONOCHROME (OPTIONAL)
                                PASS DEFINATION:
                                ----------------
                                                ONE PASS OF THE "XVSCB" CONSISTS OF ONE ITERATION OF TEST
                                                SEQUENCE.



                                EXECUTION TIME:
                                ---------------
                                                ONE PASS OF "XVSCB" ALONE ON PDP11/34 TAKES ABOUT 10SEC.



                                CONFIGURATION REQUIREMENTS:
                                ---------------------------

                                                DEFAULT PARAMETERS:
                                                DEVADR:172010,VECTOR:320,BR1:4,ICONST:1.

                                        .ENDR
```



```
                                        IDMOD   <VSCB >,172010,320,4,0,0,1
000000                                  MODULE  140000,/SCB ,172010,320,4,0,0,1,
                                        .TITLE  VSCB DEC/X11 SYSTEM EXERCISER MODULE
                                ;       ODACOM  VERSION  6       23-MAY-78
                                        .LIST   BIN
                                ;*****************************************************************
000000                          BEGIN:
000000  126     123     103     MODNAM: .ASCII  /VSCB / ;MODULE NAME.
000003  102     040
000005  000                     XFLAG:  .BYTE   OPEN            ;USED TO KEEP TRACK OF #BUFF USAGE
000006  172010                  ADDR:   172010+0                ;1ST DEVICE ADDR.
000010  000320                  VECTOR: 320+0                   ;1ST DEVICE VECTOR.
000012  200                     BR1:    .BYTE   PRTY4+0         ;1ST BR LEVEL.
000013  000                     BR2:    .BYTE   PRTY0+0         ;2ND BR LEVEL.
000014  000001                  DVID1:  0+1                     ;DEVICE INDICATOR 1.
000016  000000                  SR1:    OPEN                    ;SWITCH REGISTER 1
000020  000000                  SR2:    OPEN                    ;SWITCH REGISTER 2
000022  000000                  SR3:    OPEN                    ;SWITCH REGISTER 3
000024  000000                  SR4:    OPEN                    ;SWITCH REGISTER 4
                                ;*****************************************************************
000026  140000                  STAT:   140000                  ;STATUS WORD.
000030  001000'                 INIT:   START                   ;MODULE START ADDR.
000032  000222'                 SPOINT: MODSP                   ;MODULE STACK POINTER.
000034  000000                  PASCNT: 0                       ;PASS COUNTER.
000036  000001                  ICONT:  1                       ;# OF ITERATIONS PER PASS=1
000040  000000                  ICOUNT: 0                       ;LOC TO COUNT ITERATIONS
000042  000000                  SOFCNT: 0                       ;LOC TO SAVE TOTAL SOFT ERRORS
000044  000000                  HRDCNT: 0                       ;LOC TO SAVE TOTAL HARD ERRORS
000046  000000                  SOFPAS: 0                       ;LOC TO SAVE SOFT ERRORS PER PASS
000050  000000                  HRDPAS: 0                       ;LOC TO SAVE HARD ERRORS PER PASS
000052  000000                  SYSCNT: 0                       ;# OF SYS ERRORS ACCUMULATED
000054  000000                  RANNUM: 0                       ;HOLDS RANDOM # WHEN RAND MACRO IS CALLED
000056                          CONFIG:
000056  000000                  RES1:   0                       ;RESERVED FOR MONITOR USE
000060  000000                  RES2:   0                       ;RESERVED FOR MONITOR USE
000062  000000                  SVR0:   OPEN                    ;LOC TO SAVE R0.
000064  000000                  SVR1:   OPEN                    ;LOC TO SAVE R1.
000066  000000                  SVR2:   OPEN                    ;LOC TO SAVE R2.
000070  000000                  SVR3:   OPEN                    ;LOC TO SAVE R3.
000072  000000                  SVR4:   OPEN                    ;LOC TO SAVE R4.
000074  000000                  SVR5:   OPEN                    ;LOC TO SAVE R5.
000076  000000                  SVR6:   OPEN                    ;LOC TO SAVE R6.
000100  000000                  CSRA:   OPEN                    ;ADDR OF CURRENT CSR.
000102                          SBADR:                          ;ADDR OF GOOD DATA, OR
000102  000000                  ACSR:   OPEN                    ;CONTENTS OF CSR.
000104                          #ASADR:                         ;ADDR OF BAD DATA, OR
000104  000000                  ASTAT:  OPEN                    ;STATUS REG CONTENTS.
000106                          ERRTYP:                         ;TYPE OF ERROR
000106  000000                  ASB:    OPEN                    ;EXPECTED DATA.
000110  000000                  A#AS:   OPEN                    ;ACTUAL DATA.
000112  001026'                 RSTRT:  RESTRT                  ;RESTART ADDRESS AFTER END OF PASS
000114  000000                  #DTO:   OPEN                    ;#WORDS TO MEMORY PER ITERATION
000116  000000                  #DFR:   OPEN                    ;#WORDS FROM MEMORY PER ITERATION
000120  000000                  INTR:   OPEN                    ;# OF INTERRUPTS PER ITERATION
000122                          IDNUM:                          ;MODULE IDENTIFICATION NUMBER=
        000040                          .REPT   SPSIZ           ;MODULE STACK STARTS HERE.
```

```
                                .NLIST
                                .WORD   0
                                .LIST
                                .ENDM
     000222                     MODSP:
                                ;*****************************************************************************
  137                           ;
  138                           ; DECX MODULE FOR THE VSV-11 GRAPHICS IMAGE PROCESSER.
  139                           ;
  140                           ; NOTE THAT THIS PROGRAM REQUIRES THE MACRO PACKAGE DDXCOM.C6.
  141                           ;
  142                           .SBTTL
  143                           .SBTTL
  144                           .SBTTL  *** DECX MODULE FOR VSV-11
  145                           .SBTTL
  146                           .SBTTL
  147                           ;
  148                           ; BUILD A RUN TIME MODULE UNDER XXDP+ AS FOLLOWS:
  149                           ;
  150                           ;       .R DXCL<CR>
  151                           ;           ::
  152                           ;       BLAH-BLAH-BLAH          ; GREETINGS.
  153                           ;           ::
  154                           ;       *CNF/NP<CR>             ; CONFIGURE, NO PROMPT.
  155                           ;       *MONITOR: C<CR>         ; USE C MONITOR.
  156                           ;       *MDL VSCA<CR>           ; THIS MODULE NAME...
  157                           ;       *MDL NAM2<CR>           ;...AND ANY OTHERS.
  158                           ;           ::
  159                           ;           ::
  160                           ;       *EX<CR>                 ; LEAVE CONFIGURE MODE.
  161                           ;
  162                           ;       *LINK ODEV:NAME.BIN=IDEV:LIBNAM.LIB<CR>
  163                           ;       SYS SIZE: 1B0000<CR>
  164                           ;           ::
  165                           ;       BLAH-BLAH-BLAH          ; LINKER PROGRESS REPORTS.
  166                           ;           ::
  167                           ;       LINK DONE
  168                           ;       *
  169                           ;
  170                           ;PRGSIZ=        ^H<LASTAD>+1    ; PROGRAM SIZE IN 1/8K UNITS.
  171                           .NLIST  ME
  172                           .LIST   MEB
```



```
  174                           .SBTTL  *** VSV-11 INSTRUCTION SET.
  175                           ;
  176                           ;PLOTTING INSTRUCTIONS.
  177                           ;
  178     100000                CHAR=   100000          ;CHARACTER MODE.
  179     104000                SVEC=   104000          ;SHORT VECTOR MODE.
  180     110000                LVEC=   110000          ;LONG VECTOR MODE.
  181     114000                APNT=   114000          ;ABSOLUTE POINT MODE.
  182     120000                GHX=    120000          ;GRAPH/HISTOGRAM X MODE.
  183     124000                GHY=    124000          ;GRAPH/HISTOGRAM Y MODE.
  184     130000                RPNT=   130000          ;RELATIVE POINT MODE.
  185                           ;
  186                           ;PIXEL DATA FIELD <10:2> FOR THE ABOVE OPCODES.
  187                           ;
  188     002000                L0=     ^B10000000000           ;INTENSITY LEVEL 0 (BLACK)
  189     002000                NONE=   L0
  190     003774                ALL=    L0!<255.*4>     ; MAX IS 256. GREY SCALE LEVELS.
  191     003774                WHITE=  ALL
  192     002170                BLU=    L0!<36*4>       ; ON THE GREY-SCALE, BLUE = L36...
  193     002500                RED=    L0!<120*4>      ;...RED = L120...
  194     003140                GRN=    L0!<230*4>      ;...AND GREEN = L230.
  195                           ;
  196                           ; BIT MAP MODE DEFINITIONS.
  197                           ;
  198     136000                BM14=   136000          ; MODE 1 WITH 4 BIT PIXELS.
  199     137000                BM18=   137000          ; MODE 1 WITH 8 BIT PIXELS.
  200                                                   ; PIXEL COUNT IN <8:0>.
  201
  202     134000                BM04=   134000          ; MODE 0 WITH 4 BIT PIXELS.
  203     135000                BM08=   135000          ; MODE 0 WITH 8 BIT PIXELS.
  204     000000                M32=    0               ;  32 SQUARE PIXEL ARRAY.
  205     000001                M64=    1               ;  64 SQUARE.
  206     000002                M128=   2               ; 128 SQUARE.
  207     000003                M256=   3               ; 256 SQUARE.
  208     000010                EX2=    10              ; EXPAND BY 2, NO SMOOTHING.
  209     000020                EX4=    20              ; EXPAND BY 4, NO SMOOTHING.
  210     000050                EXSM2=  50              ; EXPAND AND SMOOTH BY 2.
  211     000060                EXSM4=  60              ; EXPAND AND SMOOTH BY 4.
  212     000100                R256=   100             ; 256 X 256 RESOLUTION.
```

```
                        ;
                        ;CONTROL AND STATUS INSTRUCTIONS.
                        ;
146040                  CUOFF=  146040          ;CURSOR (JOY-STICK) VIDEO OFF.
146060                  CUON=   146060          ;CURSOR VIDEO ON.
146100                  CURD=   146100          ;READ JSX,JSY => DXR,DYR.
146016                  CUIM=   146016          ;INTERRUPT ON MATCH (SWITCH DISABLED).
146013                  CUIS=   146013          ;INTERRUPT ON SWITCH (MATCH DISABLED).
146012                  CUIOFF= 146012          ;INTERRUPTS (BOTH) OFF.

150000                  SETHB=  150000          ;SET GRAPH/HISTO BASE LINE.
152000                  SETCB=  152000          ;SET CHARACTER BASE ADDRESS.

160000                  DJMP=   160000          ;DISPLAY JUMP.
164000                  DNOP=   164000          ;DISPLAY NO-OP.
165000                  DPOP=   165000          ;DPOP = RTS FROM CHARACTER SUB-PIX.
164001                  SYNC=   DNOP+1          ;PROCEED IN SYNC = DNOP + N <8:0>.
                                                ;WAITS FOR N OCCURRENCES OF VERTICAL...
                                                ;...SYNC START, THEN PROCEEDS...
                                                ;...(16.6MS @ 60HZ OR 20MS @ 50HZ).

000010                  ADDBAS= 10              ; DIFFERENCE BETWEEN VSV11 BASE ADDRESSES <VERSION A1>
000020                  VECBAS= 20              ; DIFFERENCE BETWEEN VECTORS            <VERSION A1>
170200                  SWTCH= 170200           ;SWITCH READ/WRITE STATUS.
170140                  CLRMEM= 170140          ;CLEAR IMAGE MEMORY TO 0'S.
170100                  SETMEM= 170100          ;SET IMAGE MEMORY TO CURRENT PIX DATA.

172000                  STOP=   172000          ;DISPLAY STOP.
171000                  SIOFF=  171000          ;STOP INTERRUPT DISABLE.
171400                  SION=   171400          ;STOP INTERRUPT ENABLE.
173000                  STOPN=  STOP!SIOFF      ;STOP, DO NOT INTERRUPT.
173400                  STOPI=  STOP!SION       ;STOP AND INTERRUPT.

174100                  GXI=    174100          ;SET X INCREMENT <5:0> FOR GRAPH/HST Y.
174100                  GYI=    GXI             ;SET Y INCREMENT <5:0> FOR GRAPH/HST X.

176000                  PROTEC= 176000          ;PROTECT MEMORY (OFF).
176050                  READ=   176040!SWE      ;READ MEM (DISPLAY), SWITCH ENABLED.
176034                  WRT=    176024!SWE      ;WRITE MEM (1'S AND 0'S), SWITCH ENABLED
176036                  WRT1=   176026!SWE      ;WRITE MEM (1'S ONLY), SWITCH ENABLED.
000010                  SWE=    10              ; ENABLE SWITCH READ/WRITE.
176074                  RDWRT=  READ!WRT        ;READ, WRITE ALL DURING RETRACE.
176076                  RDWRT1= READ!WRT1       ;READ, WRITE 1'S DURING RETRACE.

000000                  CH0=    0               ;CHANNEL SELECT BITS...
000400                  CH1=    400             ;...FOR MEMORY CONTROL...
001000                  CH2=    1000            ;...INSTRUCTIONS (176XXX).
001400                  CH3=    1400

000000                  JS0=    CH0             ;JOY-STICK UNIT SELECT BITS...
000400                  JS1=    CH1             ;...FOR CURSOR CONTROL...
001000                  JS2=    CH2             ;...INSTRUCTIONS (146XXX).
001400                  JS3=    CH3
```



```
                        ;
                        ; COLOR LOOK-UP-RAM EQUATES
                        ;
100000                  LDUN=   100000          ; CSR REGISTER -- FUNCTION DONE BIT.
010000                  LREAD=  ^B1000000000000         ; READ DATA FROM ADDRESS <7:0>.
002000                  LDI=    ^B10000000000           ; INTERRUPT ON READ/WRITE DONE.

010000                  LSET=   ^B1000000000000         ; DATA REGISTER -- SET ALL ADDRESSES...
                                                ;...WITH DATA IN <11:0>...
                                                ;...BLUE<11:8>, GREEN<7:4>, RED<3:0>.

000100                  GCOFF=  ^B1000000               ; MAINTENANCE REGISTER...
                                                ;...GAMMA CORRECTION OFF.
                        ;
                        ;OTHER USEFUL DEFINITIONS.
                        ;
040000                  I=      ^B1000000000000000              ;INTENSIFY VECTOR OR POINT.
020000                  MSX=    ^B100000000000000               ;MINUS SIGNS FOR...
000100                  MSY=    ^B1000000               ;...SHORT FORM VECTOR DATA.
020000                  MXY=    ^B100000000000000               ;MINUS SIGN FOR LONG FORM DATA.
040000                  HST=    ^B1000000000000000              ;USE GHX/GHY DATA AS HISTOGRAM.

001776                  MAXX=   511.*2          ;MAXIMUM X ADDRESS (512 X 512 X 2).
001776                  MAXY=   MAXX            ;SAME FOR Y.
000776                  HAFX=   <MAXX/2>-1      ;HALF MAX X (CENTER SCREEN).
000776                  HAFY=   HAFX            ;SAME FOR Y.

001776                  MAXY50= 511.*2          ;MAX VISIBLE Y ON 50HZ SYSTEM.
000776                  HAFY50= <MAXY50/2>-1    ;HALF VISIBLE Y.

001676                  MAXY60= <511.-32.>*2    ;MAX VISIBLE Y ON 60HZ SYSTEM.
000736                  HAFY60= <MAXY60/2>-1    ;HALF VISIBLE Y.

000002                  DX=     2               ;DELTA X (1 PIXEL UNIT).
000002                  DYI=    DX              ;DELTA Y, INTERLACED MODE.
000004                  DYNI=   DX*2            ;DELTA Y, NON-INTERLACED MODE.

000014                  SETRR= 0+14             ; SET RELOCATE REGISTER WITH <15:4>...
                                                ;...DSR<15:4> BECOMES RR<17:6>...
                                                ;...VSV RELOCATES ON 32. WORD BOUNDARY.
```

```
                                .SBTTL  *** SPECIAL MACROS AND OPDEFS.
                                ;
                                ; MACROS FOR ASSEMBLING VECTOR DATA.
                                ; CALLING ARGUMENTS ARE:
                                ;       A = INTENSIFY BIT, "I" OR "U".
                                ;       B = DELTA X, +/-0 THRU 1776 (76 IF SHORT TYPE).
                                ;       C = DELTA Y,            DITTO
                                ;
                                        .MACRO  SXY A,B,C       ;ASSEMBLE SHORT (1 WORD) DATA.
                                        $I = ^B100000000000000
                                        $X = <B*200>
                                        $Y = C
                                        .IIF IDN <U> <A>, $I = 0
                                        .IIF LT B, $X = <-B*200>!<^B100000000000000>
                                        .IIF LT C, $Y = <-C>!<^B1000000>
                                        .WORD   $I+$X+$Y
                                        .ENDM   SXY

                                        .MACRO  LXY A,B,C       ;ASSEMBLE LONG (2 WORDS) DATA.
                                        $I = ^B100000000000000
                                        $X = B
                                        $Y = C
                                        .IIF IDN <U> <A>, $I = 0
                                        .IIF LT B, $X = <-B>!<^B100000000000000>
                                        .IIF LT C, $Y = <-C>!<^B100000000000000>
                                        .WORD   $I+$X, $Y
                                        .ENDM   LXY
                                ;
                                ;SOME HANDY -11 OPDEF'S.
                                ;
000401                          SKP1=   BR+1                    ;SKIP NEXT WORD.
000402                          SKP2=   BR+2                    ;SKIP NEXT 2 WORDS.
000403                          SKP3=   BR+3                    ;SKIP NEXT 3 WORDS.
```



```
                                .SBTTL  *** DEVICE REGISTER ASSIGNMENTS
                                ;
                                ; DEVICE REGISTER ASSIGNMENTS.
                                ;
000222  172010                 DPC:    172010          ;DISPLAY PC
000224                          DRR:
000224  172012                 DSR:    172012          ;DISPLAY STATUS REGISTER (ALSO REG REG).
000226  172014                 DXR:    172014          ;DISPLAY X POSITION REGISTER.
000230  172016                 DYR:    172016          ;DISPLAY Y POSITION REGISTER.

000232  172020                 LSR:    172020          ; LOOK-UP-RAM CSR.
000234  172022                 LDR:    172022          ; LOOK-UP-RAM DATA REGISTER.
000236  172024                 LMR:    172024          ; LOOK-UP-RAM MAINTENANCE REGISTER.

000240  000320                 STPV:   320             ;STOP VECTOR
000242  000324                 JSMV:   324             ;JOY-STICK MATCH VECTOR.
000244  000330                 TOTV:   330             ;TIME-OUT VECTOR
000246  000334                 JSSV:   334             ;JOY-STICK SWITCH (DEMAND) VECTOR.
000250  000340                 LUTV:   340             ; LUT READ/WRITE DONE VECTOR.
                                ; MISCELLANEOUS STORAGE REGISTERS.
                                ;
000252  000000                 SDPC:   0               ; SAVED DPC ON INTERRUPT.
000254  000000                 SDSR:   0               ;       DITTO DSR
000256  000000                 SDXR:   0               ;       DITTO DXR
000260  000000                 SDYR:   0               ;       DITTO DYR

000262  000000                 LUTFLG: 0               ; +1 IF LUT (A025) INSTALLED.

000264  000000                 SLSR:   0               ; SAVED LUT STATUS.
000266  000000                 SLDR:   0               ;       DITTO DATA.
000270  000000                 SLMR:   0               ;       DITTO MAINT.

000272  000000                 FIDX:   0               ; FRAME INDEX COUNTER (FOR PART 1).
000274  000010                 FCD:    8.              ; FRAME CHANGE DELAY COUNT...           <VERSION A1>
000276  000010                 FCD1:   8.              ;...IS DECREMENTED HERE.                <VERSION A1>
000300  000000                 ADDRX:  0               ; CURRENT BASE ADDRESS                  <VERSION A1>
000302  000000                 VECIXX: 0               ; CURRENT VECTOR ADDRESS                <VERSION A1>
000304  000000                 DEVCN:  0               ; CURRENT DEVICE CHECK MASK             <VERSION A1>
000306  000000                 DIVCHK: 0               ; USER SPECIFIED DEVICE MASK            <VERSION A1>

000310  000000                 T0:     0               ; SOME TEMP REGISTERS.
000312  000000                 T1:     0
000314  000000                 T2:     0
000316  000000                 T3:     0

000320  003726'                VADDR:  DBEGIN          ; DISPLAY CODE VIRTUAL ADDRESS.
000322  000000                 PADDR:  0               ; CONVERTED TO PHYSICAL HERE...
000324  000000                 EABITS: 0               ;...WITH EA BITS HERE <5:4>.
000326  000000                 DSTRT:  0               ; DISPLAY START (CLEAR) DISPATCHER.
000330  000000                 DLOOP:  0               ; DISPLAY LOOPING DISPATCHER.
000332  000014                 RELOC:  SETRR           ; PLUS RELOCATION BITS IN <15:4>.
000334  000000                 RELOK:  0               ; NON-ZERO (CAN'T RUN) IF RELOCATION...
                                                       ;...BOUNDARY IS NOT A MULTIPLE...
                                                       ;...OF 100B (32. WORDS).
000336  000000                 BADPAD: 0               ; PHYSICAL ADDRESS WHICH CAUSED...
                                                       ;..."RELOK" TO BE NON-ZERO.
```

```
400 000340  000000                  TEMPS:  0               ;STORE TEMP PS
401 000342  000000                  TEMPC:  0               ;STORE TEMP PC
402         000004                  MEMERR=4                ;NON EXIST MEM TRAP
```



```
404                                 .SBTTL  *** JOY-STICK AND TIME-OUT INTERRUPT SERVICE ROUTINES.
405                                 ;
406                                 ; JOY-STICK INTERRUPT SERVICE.
407                                 ;
408 000344                          MATCH:
409 000344                          SWITCH:
    000344  000004  000000' 000352'        PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
410 000352  004767  000074          1$:    JSR     PC,SAVDPU       ; SAVE REGISTERS...
411 000356  000240  000240                 240,240                 ;...AND THATS ALL (FOR NOW).
412                                        ;
413                                        ;
414                                        ;
415                                        ;
416 000362  000167  001424                 JMP     RESM            ; RESUME AND EXIT.
417                                 ;
418                                 ; DISPLAY TIME-OUT MEANS TROUBLE -- QUIT.
419                                 ;
420 000366                          TIMOUT:
    000366  000004  000000' 000374'        PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
421 000374  004767  000052          1$:    JSR     PC,SAVDPU       ; SAVE REGISTERS.
422 000400  104401  000000' 000420'        MSG$,BEGIN,2$   ;ASCII MESSAGE CALL
423 000406  046767  177672  177702        BIC     DEVON,F3        ; CLR THIS ONE FROM MASK        <<VER B>>
424 000414  000167  000406                 JMP     RESINT          ; GET THE NEXT ONE
425
426 000420      045     040     040 2$:    .ASCIZ  '%  VSCA0 -- DPU TIME-OUT%'
    000423      126     123     103
    000426      101     060     040
    000431      055     055     040
    000434      104     120     125
    000437      040     124     111
    000442      115     105     055
    000445      117     125     124
    000450      045     000
427                                        .EVEN
428                                 ;
429                                 ; COMMON ROUTINE TO SAVE DPU REGISTERS.
430                                 ;
431 000452  017767  177544  177572  SAVDPU: MOV     @#DPC,SDPC      ; SAVE ALL DPU REGISTERS.
432 000460  017767  177540  177566          MOV     @DSR,SDSR
433 000466  017767  177534  177562          MOV     @DXR,SDXR
434 000474  017767  177530  177556          MOV     @DYR,SDYR
435 000502  000240                          240
436 000504  000240                          240             ; AND ANYTHING ELSE WE MIGHT NEED.
437 000506  000240                          240
438 000510  000207                          RTS     PC
```

```
                                        .SBTTL  *** LOOK-UP-TABLE SERVICE ROUTINES.
                                        ;
                                        ; NORMALLY, WE DON'T EXPECT ANY LUT INTERRUPTS.
                                        ; REPORT AND RESTART IF WE GET ONE.
                                        ;
000512                                  UNXLI:
000512  000004  000000' 000520'             PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
000520  017767  177506  177536  1$:       MOV     @LSR,SLSR       ; SAVE LUT REGISTERS.
000526  017767  177502  177532            MOV     @LDR,SLDR
000534  017767  177476  177526            MOV     @LMR,SLMR
000542  000240  000240                     240,240                 ; IN CASE WE NEED SOMETHING ELSE
000546  104401  000000' 000560'             MSG$,BEGIN,2$   ;ASCII MESSAGE CALL
000554  000167  000246                     JMP     RESTRT          ;...AND RESTART.

000560      045     040     040  2$:       .ASCIZ  '% VSCA0 -- UNEXPECTED LUT INTERRUPT%'
000563      126     123     103
000566      101     060     040
000571      055     055     040
000574      125     116     105
000577      130     120     105
000602      103     124     105
000605      104     040     114
000610      125     124     040
000613      111     116     124
000616      105     122     122
000621      125     120     124
000624      045     000
                                        .EVEN
                                        ;
                                        ; BLAST A 256 LEVEL GREY-SCALE FROM THE DATA IN "LUMTBL".
                                        ;
000626  012777  000656' 177414  GRASCL: MOV     #2$,@LUTV       ; SET DONE VECTOR.
000634  012777  002000  177370          MOV     #LD1!0,@LSR     ; INTERRUPT AND ADDR 0.
000642  012700  015112'                 MOV     #LUMTBL,R0      ; TABLE ADDRESS.
000646  012077  177362          1$:     MOV     (R0)+,@LDR      ; FILL THE LUT RAM.
000652  104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
000656                                  2$:
000656  000004  000000' 000664'             PIRQ$,BEGIN,3$          ; QUEUE UP TO CONTINUE AT 3$ AND RTI
000664  105777  177342          3$:     TSTB    @LSR            ; LUT RAM FULL ??
000670  001366                          BNE     1$              ; NO, LOOP.
000672  000431                          BR      BLXIT           ; YES, RETURN.
                                        ;
                                        ; BLAST 16 SHADES OF THE BASIC COLOR IN (R0).
                                        ;
000674  012777  000732' 177346  SHADES: MOV     #3$,@LUTV       ; SET WRITE DONE VECTOR.
000702  012777  002000  177322          MOV     #LD1!0,@LSR     ; ADDR 0, INTERRUPT WHEN DONE.
000710  042700  177356                  BIC     #^C421,R0       ; SET DIMMEST SHADE THIS COLOR.
000714  005001                          CLR     R1              ; 1ST SHADE IS ALWAYS BLACK.
000716  012702  000020          1$:     MOV     #16.,R2         ; 16 WORDS / SHADE.
000722  010177  177306          2$:     MOV     R1,@LDR         ; WRITE A WORD.
000726  104400  000000'                 EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
000732                                  3$:
000732  000004  000000' 000740'             PIRQ$,BEGIN,4$          ; QUEUE UP TO CONTINUE AT 4$ AND RTI
000740  105777  177266          4$:     TSTB    @LSR            ; LUT RAM FULL ??
000744  001404                          BEQ     5$              ; EXIT IF SO.
000746  005302                          DEC     R2              ; THIS SHADE DONE ??
000750  001364                          BNE     2$              ; NOT YET, LOOP.
```



```
000752  060001                          ADD     R0,R1           ; YES, INCREMENT THE SHADE...
000754  000760                          BR      1$              ;...AND LOOP.
000756                                  5$:
                                        ;
                                        ; COMMON EXIT FROM EITHER PATTERN.
                                        ;
000756  005077  177250          BLXIT:  CLR     @LSR            ; CLEAR STATUS...
000762  012777  000100  177246          MOV     #GCOFF,@LMR     ;...GAMMA CORRECT OFF...
000770  012777  000512' 177252          MOV     #UNXLI,@LUTV    ;...RESET VECTOR...
000776  000207                          RTS     PC              ;...AND RETURN.
```

```
                                        .SBTTL  *** PROGRAM START-RESTART
                                        ;
                                        ; INITIALIZE DISPLAY ADDRESSES, AND VECTORS.
                                        ; BLAST THE LUT (IF IT'S THERE).
                                        ;
001000                                  START:
001000  032767  000001  177012                  BIT     #1,SR2          ; USING COLOUR LOOKUP                   <VERSION A1>
001006  001404                                  BEQ     1$              ; NO                                    <VERSION A1>
001010  012767  000001  177270                  MOV     #1,DIVCHK       ; YES- FORCE ONE DEVICE                 <VERSION A1>
001016  000411                                  BR      ONE             ;                                       <VERSION A1>
001020  016767  176770  177270  1$:     MOV     DVID1,T3        ; GET MASK AND SAVE IT                  <<VER B>>
                                        ;
001026                                  RESTRT:
001026  016767  177264  177252                  MOV     T3,DIVCHK       ; PUT MASK INTO WORK LOCATION           <VERSION A1>
001034  016767  177256  177252                  MOV     T3,T2           ; ALSO INTO T2 COUNTER                  <<VER B>>
001042                                  ONE:                            ;
001042  012767  000001  177234                  MOV     #1,DEVON        ; START CHECK AT DEVICE 1               <VERSION A1>
001050  016767  176732  177222                  MOV     ADDR,ADDRX      ; SET DEVICE START ADDRESS              <VERSION A1>
001056  016767  176726  177216                  MOV     VECTOR,VECTXX   ; AND VECTOR START                      <VERSION A1>
                                        ;
001064  005767  177216          RESTRA: TST     DIVCHK          ; ANY DEVICE TO TEST                    <VERSION A1>
001070  001002                                  BNE     77$             ; YES                                   <VERSION A1>
001072  104410  000000'                        ENDS,BEGIN              ;
001076  005767  177202          77$:    TST     DEVON           ; ALL MODULES TESTED                    <VERSION A1>
001102  001736                                  BEQ     START           ; YES RESTART FROM THE TOP              <VERSION A1>
001104  036767  177174  177174                  BIT     DEVON,DIVCHK    ; IS THIS MODULE ON                     <VERSION A1>
001112  001012                                  BNE     76$             ; YES USE IT                            <VERSION A1>
001114  062767  000010  177156                  ADD     #ADDRAS,ADDRX   ; ELSE STEP TO NEXT ONE                 <VERSION A1>
001122  062767  000020  177152                  ADD     #VECBAS,VECTXX  ; AND VECTOR                            <VERSION A1>
001130  006367  177150                          ASL     DEVON           ;                                       <VERSION A1>
001134  001721                                  BEQ     START           ; IF ALL DONE RESET TO TOP              <VERSION A1>
001136  000757                                  BR      77$             ;                                       <VERSION A1>
001140                                  76$:                            ;
001140  016700  177134                          MOV     ADDRX,R0        ; GET BASE DEVICE ADDRESS.              <VERSION A1>
001144  012701  000222'                        MOV     #DPC,R1         ;
001150  010021                          1$:     MOV     R0,(R1)+        ; SET REGISTER ADDRESSES.
001152  062700  000002                          ADD     #2,R0           ;
001156  020127  000240'                        CMP     R1,#STPV        ; END OF ADDRESS BLOCK ??
001162  001372                                  BNE     1$              ; BR IF NOT.
                                        ;
001164  016700  177112                          MOV     VECTXX,R0       ; YES, SET VECTOR ADDRESSES.            <VERSION A1>
001170  010021                          2$:     MOV     R0,(R1)+        ;
001172  062700  000004                          ADD     #4,R0           ;
001176  020127  000252'                        CMP     R1,#LUTV+2      ;
001202  001372                                  BNE     2$              ; NO
                                        ;
001204  016700  177030                          MOV     STPV,R0         ; YES, SET DPU VECTORS...
001210  116701  176576                          MOVB    BR1,R1          ;...AND PRIORITIES.
001214  042701  177437                          BIC     #^C340,R1       ;
001220  012720  002022'                        MOV     #STOP1,(R0)+    ; SET STOP...
001224  010120                                  MOV     R1,(R0)+        ;
001226  012720  000344'                        MOV     #MATCH,(R0)+    ;...MATCH...
001232  010120                                  MOV     R1,(R0)+        ;
001234  012720  000366'                        MOV     #TIMOUT,(R0)+   ;...TIME-OUT...
001240  010120                                  MOV     R1,(R0)+        ;
001242  012720  000344'                        MOV     #SWITCH,(R0)+   ;...AND SWITCH VECTORS.
001246  010120                                  MOV     R1,(R0)+        ;
```



```
001250  012720  000512'                        MOV     #UNXLI,(R0)+    ; SET LUT DONE VECTOR.
001254  010120                                  MOV     R1,(R0)+        ;
                                        ;
001256  005227                          LUTIN:  INC     (PC)+           ; AUTO-SIZE FOR LUT INSTALLED.
001260  177777                                  -1                      ; ONCE-ONLY SWITCH.
001262  001040                                  BNE     1$
001264  013767  000004  177050                  MOV     @#MEMERR,TEMPC  ;STORE ADDRESS AT NON EXISTANCE
                                                                        ;MEMORY TRAP (TEMPARARY)
001272  013767  000006  177040                  MOV     @#MEMERR+2,TEMPS        ;STORE PS
001300  012737  003362' 000004                  MOV     #TRP1,@#MEMERR  ;NON EXIST MEM TRAP HERE
001306  013737  177776  000006                  MOV     @#PS,   @#MEMERR+2      ;GET CURRENT PS
001314  042737  177437  000006                  BIC     #177437,@#MEMERR+2      ;CLEAR UNWANTED BITS
001322  005067  176734                          CLR     LUTFLG          ; ASSUME NO LUT INSTALLED.
001326  032767  000001  176464                  BIT     #1,SR2          ; USING COLOUR LOOKUP                   <VERSION A1>
001334  001405                                  BEQ     3$              ;                                       <VERSION A1>
001336  005777  176670                          TST     @LSR            ; TRAP AND RESTART IF NO LUT.
001342  012767  000001  176712                  MOV     #1,LUTFLG       ; OTHERWISE, SET LUT AVAILABLE.
001350                                  3$:                             ;                                       <VERSION A1>
001350  016737  176766  000004                  MOV     TEMPC,@#MEMERR  ;RECOVER ERRVEC PC
001356  016737  176756  000006                  MOV     TEMPS,@#MEMERR+2        ;
                                        ;
001364                                  1$:
                                        ;
```

```
                                        ;
                                        ; NOW COMPUTE ANY RELOCATION OFFSET BASED ON:
                                        ;       OFFSET = PHYSICAL - VIRTUAL
                                        ;
                                        ; VSV RELOCATES ON 32. WORD (100 BYTES OCTAL) BOUNDARIES,
                                        ; IN THE TRADITIONAL KT STYLE.
                                        ; DURING PROGRAM RELOCATION (IF ENABLED), PA<5:0> AND VA<5:0>
                                        ; SHOULD BE IDENTICAL.  IF NOT, THE RELOCATION PROCESS DIDN'T
                                        ; WORK RIGHT.  TELL THE MAN, BUT DON'T TRY TO RUN ANYTHING.
001364                                  SETUP:
001364 012767  004154' 010730                   MOV     #SUB.1,ACAT+2   ; SET SUBROUTINE CALLS...
001372 012767  005132' 010724                   MOV     #SUB.2,ACAT+4   ;...IN VIRTUAL CHAR...
001400 012767  012270' 010720                   MOV     #SUB.3,ACAT+6   ;...ADDRESS TABLE.
001406 012767  003726' 176704                   MOV     #DBEGIN,VADDR   ; CONVERT VIRTUAL TO PHYSICAL.
001414 104415  000000' 000320'                   GETPAS,BEGIN, VADDR             ;GET PHYSICAL ADDRESS FROM 16-BIT VADDR
001422 016701  176674                           MOV     PADDR,R1
001426 042701  000077                           BIC     #77,R1          ; PA<15:6> => R1.
001432 016700  176666                           MOV     BABITS,R0       ; PA<17:16> => R0<5:4>.
001436 000300                                   SWAB    R0
001440 006100                                    ROL    R0
001442 006100                                     ROL   R0              ; PA<17:16> => R0<15:14>.
001444 006001                                   ROR     R1
001446 006001                                    ROR    R1
001450 050001                                     BIS   R0,R1           ; PA<17:6> => R1<15:4>.
001452 016700  176642                           MOV     VADDR,R0
001456 042700  000077                           BIC     #77,R0          ; VA<15:6> => R0.
001462 000241                                   CLC
001464 006000                                    ROR    R0
001466 006000                                     ROR   R0              ; VA<15:6> => R0<13:4>.
001470 160001                                   SUB     R0,R1           ; PHY - VIRT = RELOC.
001472 052701  000014                           BIS     #SETRR,R1       ; MAKE SETRR INSTRUCTION...
001476 010167  176630                           MOV     R1,RELOC        ;...FOR DSR (RR REGISTER)...
001502 016767  176612  176616                   MOV     VADDR,DSTRT     ;...AND VIRTUAL START FOR DPC.

001510 005067  176620                           CLR     RELOK           ; NOW, ASSUME RELOC IS VALID.
001514 016700  176602                           MOV     PADDR,R0
001520 166700  176574                           SUB     VADDR,R0
001524 042700  177700                           BIC     #^C77,R0        ; PA<5:0> = VA<5:0> ???
001530 001414                                   BEQ     1$              ; WE'RE OK IF SO.
001532 005167  17657b                           COM     RELOK           ; NO, INHIBIT START-UP.
001536 026767  176560  176572                   CMP     PADDR,BADPAD    ; SAME PA AS LAST TIME ??
001544 001406                                   BEQ     1$              ; YES, DON'T REPORT AGAIN.
001546 016767  176550  176562                   MOV     PADDR,BADPAD    ; NO, SAVE THE BAD NUMBER...
001554 104401  000000' 001564'                   MSGS,BEGIN,CANT ;ASCII MESSAGE CALL
001562 000423                           1$:     BR      PART1           ;...AND FALL THRU.
                                                                        ; "RELOK" NON-ZERO WILL CAUSE...
                                                                        ;...US TO "ENDIT".  SOONER OR...
                                                                        ;...LATER, WE'LL RESTART AND...
                                                                        ;...GET A NEW RELOCATION VALUE.

001564      045     040     040 CANT:   .ASCIZ  '%  VSCA0 -- BAD RELOCATION BOUNDARY%'
001567      126     123     103
001572      101     060     040
001575      055     055     040
001600      102     101     104
001603      040     122     105
001606      114     117     103
```



```
001611      101     124     111
001614      117     116     040
001617      102     117     125
001622      116     104     101
001625      122     131     045
001630      000
                                        .EVEN
```

```
                                        .SBTTL  *** PART 1 -- THE STANDARD VISUAL REFERENCE FRAME
                                        ;
                                        ; CLEAR SCREEN, BLAST LUT (OPTIONAL) AND START DISPLAY.
                                        ;
001632  005767  176476          PART1:  TST     RELOC
001636  001402                          BEQ     1$              ; *** NOP TO DEBUG ***
001640  000167  000620                  JMP     DONE2           ; RELOC INVALID, CAN'T RUN.
                                        
001644  013767  000004  176470  1$:     MOV     @#MEMERR,TEMPC  ;STORE ADDRESS AT NON EXISTANCE
                                                                ;MEMORY TRAP (TEMPARARY)
001652  013767  000006  176460          MOV     @#MEMERR+2,TEMPS        ;STORE PS
001660  012737  003404' 000004          MOV     #TRP2,@#MEMERR  ;NON EXIST MEM TRAP HERE
001666  016777  176440  176330          MOV     RELOC,@DSR      ; SET RR REGISTER.
001674  016777  176426  176320          MOV     DSTRT,@DPC      ; CLEAR SCREEN RAMS.
001702                          2$:
        001702  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
        001706  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
001712  005777  176306                  TST     @DSR
001716  100371                          BPL     2$
001720  005767  176336                  TST     LUTFLG
001724  001402                          BEQ     3$
001726  004767  176674                  JSR     PC,GRASCL       ; BLAST THE LUT.
001732  016737  176404  000004  3$:     MOV     TEMPC,@#MEMERR  ;RECOVER ERRVEC PC
001740  016737  176374  000006          MOV     TEMPS,@#MEMERR+2        ;RECOVER PS
001746  012777  002022' 176264          MOV     #STOP1,@STPV    ; SET DPU STOP VECTOR.
001754  016767  176346  176346          MOV     DSTRT,DLOOP
001762  062767  000020  176340          ADD     #LOGO-DBEGIN,DLOOP ; SET 1ST FRAME ADDRESS.
001770  005067  176276                  CLR     FIDX            ; INIT FRAME INDEX.
                                        
001774  016767  176274  176274  NEXT:   MOV     FCD,FCD1        ; SET/RESET FRAME CHANGE DELAY.
002002  016777  176322  176212          MOV     DLOOP,@DPC      ; START/RESTART DPU...
002010  000402                          SKP2                    ;...OR...
002012  005277  176204          RESM:   INC     @DPC            ;...RESUME.
002016  104400  000000'                 EXITS,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
                                        ;
                                        ; RETURN HERE ON STOP INTERRUPT.
                                        ; CHANGE FRAMES AFTER A SUITABLE DELAY.
                                        ;
002022                          STOP1:
        002022  000004  000000' 002030'         PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
002030  004767  176416          1$:     JSR     PC,SAVDPU       ; SAVE DPU REGISTERS.
002034  005367  176236                  DEC     FCD1            ; CHANGE DELAY EXPIRED ??
002040  001360                          BNE     NEXT+6          ; NOT YET, LOOP ON THIS FRAME.
002042  062767  000002  176222          ADD     #2,FIDX         ; YES, BUMP INDEX...
002050  016700  176216                  MOV     FIDX,R0
002054  000170  002060'                 JMP     @2$(R0)         ;...AND CHANGE FRAMES.
002060  001632'                 2$:     PART1                   ; FRAME 1 (CLEAR AND LOGO).
002062  002076'                         FR2                     ;       2 (OCTAGONS).
002064  002106'                         FR3                     ;       3 (GRAPH/HISTO Y).
002066  002116'                         FR4                     ;       4 (GRAPH/HISTO X).
002070  002130'                         FR5                     ;       5 (BIT MAPS).
002072  002224'                         FR6                     ;       6 (COLOR BARS).
002074  002354'                         HUES                    ; CHANGE COLORS (OPTIONAL).
```



```
                                        ;
                                        ; SET UP SEQUENCE FOR EACH FRAME.
                                        ;
002076  062767  000110  176224  FR2:    ADD     #OCTGNS-LOGO,DLOOP ; ADJUST TO FRAME 2.
002104  000403                          SKP3
002106  062767  000140  176214  FR3:    ADD     #GRY-OCTGNS,DLOOP ; ADJUST TO FRAME 3.
002114  000403                          SKP3
002116  062767  000310  176204  FR4:    ADD     #GRX-GRY,DLOOP  ; ADJUST TO FRAME 4.
002124  000167  177644                  JMP     NEXT
                                        
002130  012700  005132'          FR5:    MOV     #OBUFR,R0       ; FRAME 5 REQUIRES THE BUFFER.
002134  012701  000400                  MOV     #256.,R1        ; SET UP TO FILL IT.
002140  012702  177777                  MOV     #-1,R2
002144  000407                          BR      2$
002146  032701  000077          1$:     BIT     #77,R1          ; BREAK EVERY 64 TIMES...
002152  001004                          BNE     2$              ;...SO WE DON'T HOG THE CPU.
002154  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
        002160  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002164  010220                  2$:     MOV     R2,(R0)+        ; FILL BIT MAP BUFFER WITH...
002166  005020                          CLR     (R0)+           ;...ALTERNATE 1'S AND 0'S.
002170  005301                          DEC     R1
002172  001365                          BNE     1$              ; LOOP 'TIL DONE.
002174  016767  176074  176074          MOV     FCD,FCD1        ; ADJUST FRAME DELAY...
002202  006267  176070                  ASR     FCD1            ;...'CAUSE BIT MAPS...
002206  006267  176064                  ASR     FCD1            ;...TAKE TOO LONG !!
002212  062767  000310  176110          ADD     #814-GRX,DLOOP  ; ADJUST ADDRESS TO FRAME 5.
002220  000167  177556                  JMP     NEXT+6
                                        
002224  012700  005132'          FR6:    MOV     #OBUFR,R0       ; FRAME 6 REQUIRES THE BUFFER.
002230  012701  112000                  MOV     #LVEC!LO,R1
002234  012702  000500                  MOV     #500,R2         ; +500 Y VALUE.
002240  012703  020500                  MOV     #-#XY!500,R3    ; -500 Y VALUE.
002244  000407                          BR      2$
002246  032701  000077          1$:     BIT     #77,R1          ; BREAK ONCE IN A WHILE...
002252  001004                          BNE     2$              ;...LIKE BEFORE.
002254  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
        002260  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002264  010120                  2$:     MOV     R1,(R0)+        ; SET VECTOR!L...
002266  012720  040000                  MOV     #I!0,(R0)+      ;...X...
002272  010220                          MOV     R2,(R0)+        ;...+Y.
002274  012720  000002                  MOV     #2,(R0)+
002300  005020                          CLR     (R0)+           ; MOVE OVER 1...
002302  062701  000004                  ADD     #4,R1           ;...AND INCREMENT THE LEVEL.
002306  010120                          MOV     R1,(R0)+        ; SET VECTOR!L+4
002310  012720  040000                  MOV     #I!0,(R0)+      ;...X...
002314  010320                          MOV     R3,(R0)+        ;...-Y...
002316  012720  000002                  MOV     #2,(R0)+
002322  005020                          CLR     (R0)+           ; MOVE OVER...
002324  062701  000004                  ADD     #4,R1           ;...AND INCREMENT LEVEL AGAIN.
002330  032701  001777                  BIT     #1777,R1
002334  001344                          BNE     1$              ; LOOP 'TIL ALL LEVELS SET.
002336  012720  165000                  MOV     #DPOP,(R0)+     ; THEN, TERMINATE THE BUFFER.
002342  062767  000060  175760          ADD     #BARS-814,DLOOP ; ADJUST TO FRAME 6 ADDRESS.
002350  000167  177420                  JMP     NEXT            ;...AND DO IT.
```

```
                                        ;
                                        ; NOW IF LUT IS THERE, REBLAST IT TO DISPLAY
                                        ; 16 SHADES OF EACH OF THE NTSC BASIC COLORS.
                                        ; DISPLAY IS STATIC (STOPPED) DURING THIS PHASE.
                                        ;
002354  005767  175702          HUES:   TST     LUTFLG
002360  001435                         BEQ     DONE1           ; BYPASS IF NO LUT AVAILABLE.
002362  012705  002434'                 MOV     #3$,R5          ; COLOR TABLE ADDRESS.
002366  012500                  1$:     MOV     (R5)+,R0        ; 1ST/NEXT COLOR.
002370  001431                         BEQ     DONE1           ; BR IF END OF TABLE.
002372  004767  176276                 JSR     PC,SHADES       ; BLAST IT.
002376  016767  175672  175672         MOV     FCD,FCD1
002404  006367  175666                 ASL     FCD1            ; EXTEND THE DELAY TIME.
002410  006367  175662                 ASL     FCD1
002414                          2$:
002414  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
002420  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002424  005367  175646                 DEC     FCD1
002430  001371                         BNE     2$              ; DELAY...
002432  000755                         BR      1$              ;...AND LOOP THRU ALL COLORS.

002434  007400                  3$:     7400            ; BLUE
002436  000017                          0017            ; RED
002440  007417                          7417            ; MAGENTA
002442  000360                          0360            ; GREEN
002444  007760                          7760            ; CYAN
002446  000377                          0377            ; YELLOW
002450  007777                          7777            ; WHITE
002452  000000                          0               ; BLACK (END)
                                        ;
                                        ; THIS IS A COMMON "END-ITERATION" POINT FOR BOTH PARTS.
                                        ;
002454  032767  000001  175334  DONE1:  BIT     #BIT0,SR1       ; PART 2 ENABLED ??
002462  001031                         BNE     PART2           ; DO IT IF  SO.
002464  046767  175614  175622  DONE2:  BIC     DEVON,T2        ; DONE ALL MODULES?             <<VER B>>
002472  001421                         BEQ     PASS            ; YES SIGNAL EOP                <<VER B>>
002474  006367  175604                 ASL     DEVON           ; SELECT NEXT MODULE            <VERSION A1>
002500  032767  000001  175312         BIT     #1,SR2          ; USING COLOR LOOKUP?           <<VER B>>
002506  001403                         BEQ     2$              ; NO!                           <<VER B>>
002510  062767  000010  175562         ADD     #10,ADDRX       ; FOUR MORE CSR'S THEN          <<VER B>>
002516  062767  000010  175554  2$:     ADD     #ADDBAS,ADDRX   ; GET NEW ADDRESS               <VERSION A1>
002524  062767  000020  175550         ADD     #VECBAS,VECTXX  ; PLUS VECTOR                   <VERSION A1>
002532  000167  176326                 JMP     RESTRA          ; DO NEXT ONE

002536                          PASS:
002536  104413  000000'                 END1T$,BEGIN            ;SIGNAL END OF ITERATION.
002542  000167  176260                 JMP     RESTRT          ; GO FOR MORE                   <<VER B>>
```



```
                                        .SBTTL  *** PART 2 -- THE OLD BOUNCING BALLS
                                        .SBTTL  ***     SR1 BIT0 = 1 = ENABLE PART 2.
                                        .SBTTL  ***     SR1 BIT1 = 1 = USE *PITE-IN-RETRACE MODE.
                                        .SBTTL  ***     SR1 BIT2 = 1 = DISABLE BELL ON BOUNCE.
                                        .SBTTL
                                        ;
                                        ; DISPLAY TWO OCTAGONS FLOATING AROUND AT RANDOM.
                                        ; BOUNCE THEM OFF THE *ALLS AND "CRASH" IF THEY TOUCH.
                                        ;
002546  016777  175554  175446  PART2:  MOV     DSTRT,@DPC      ; CLEAR SCREEN (R/W MODE).
002554                          1$:
002554  104407  000000'                 BREAK$,BEGIN            ;TEMPORARY RETURN TO MONITOR....
002560  104407  000000'                 BREAK$,BEGIN            ;THEN CONTINUE AT NEXT INSTRUCTION.
002564  005777  175434                 TST     @DSR
002570  100371                         BPL     1$
002572  005767  175464                 TST     LUTFLG
002576  001402                         BEQ     2$
002600  004767  176022                 JSR     PC,GRASCL
002604  012767  002000  175500  2$:     MOV     #2000,T1        ; SETUP DEAD MAN TIMER          <<VER B>>
002612  012767  000200  000536         MOV     #200,LOLIM
002620  012767  001600  000532         MOV     #1600,HILIM     ;SET LIMITS FOR ORIGINS.
002626  004767  000456                 JSR     PC,RAND
002632  010167  007340                 MOV     R1,RB           ;NEW ORIGIN, RED BALL.
002636  004767  000446                 JSR     PC,RAND
002642  010167  007332                 MOV     R1,RB+2
002646  004767  000436                 JSR     PC,RAND
002652  010167  007332                 MOV     R1,GB           ;NEW ORIGIN, GREEN BALL.
002656  004767  000426                 JSR     PC,RAND
002662  010167  007324                 MOV     R1,GB+2
002666  012767  000004  000462         MOV     #4,LOLIM
002674  012767  000030  000456         MOV     #30,HILIM       ;SET LIMITS FOR DELTAS.
002702  004767  000402                 JSR     PC,RAND
002706  010167  000624                 MOV     R1,DX1          ;BALL 1, DELTA X...
002712  004767  000372                 JSR     PC,RAND
002716  010167  000616                 MOV     R1,DX1+2        ;...AND Y.
002722  004767  000362                 JSR     PC,RAND
002726  010167  000610                 MOV     R1,DX2          ;BALL 2, DITTO.
002732  004767  000352                 JSR     PC,RAND
002736  010167  000602                 MOV     R1,DX2+2

002742  016767  175360  175360         MOV     DSTRT,DLOOP
002750  062767  006210  175352         ADD     #B1-DBEGIN,DLOOP ; INIT STARTING ADDRESS.
002756  032767  000002  175032         BIT     #BIT1,SR1       ; DYNAMIC MODE (DEFAULT) ??
002764  001007                         BNE     3$              ; BR IF NOT SO.
002766  062767  000012  175334         ADD     #B1A-B1,DLOOP   ; NO, ADJUST STARTING ADDRESS...
002774  012767  164002  007256         MOV     #SYNC+1,B4      ;...AND OPCODE FOR R/W MODE;
003002  000414                         BR      4$
003004  016777  175320  175210  3$:     MOV     DLOOP,@DPC      ; DYNAMIC, RECONFIGURE IM'S.
003012  005777  175206                 TST     @DSR            ; WAIT FOR DPU STOP.
003016  100375                         BPL     .-4
003020  062767  000036  175302         ADD     #B2-B1,DLOOP    ; ADJUST STARTING ADDRESS...
003026  012767  170340  007224         MOV     #SWTCH!CLRMEM,B4 ;...AND OPCODE FOR DYNAM MODE.
```

```
828 003034  012767  020040  007160  4$:     MOV     #20040,B3       ; NULL (SPACE) THE CRASH TEXT.
829 003042  016767  007154  007154          MOV     B3,B3+2
830 003050  016767  007146  007150          MOV     B3,B3+4
831 003056  012777  003114' 175154          MOV     #STOP2,@STPV    ; SET STOP VECTOR...
832 003064  016767  175204  175204          MOV     FCD,FCD1
833 003072  006367  175200                   ASL     FCD1            ;...EXTEND THE FRAME DELAY.
834 003076  006367  175174                   ASL     FCD1
835 003102  016777  175222  175112  CNTNU:  MOV     DLOOP,@DPC      ; START/RESTART DISPLAY.
836 003110  104400  000000'                  EXIT$,BEGIN                     ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
837                                 ;
838                                 ; CONTINUE HERE ON STOP INTERRUPT.
839                                 ; MOVE THE BALLS AROUND UNTIL THEY CRASH, THEN END-PASS.
840                                 ;
841 003114                          STOP2:
    003114  000004  000000' 003122'         PIRQ$,BEGIN,1$          ; QUEUE UP TO CONTINUE AT 1$ AND RTI
842 003122  126727  007074  000040  1$:     CMPB    B3,#40          ; ARE WE STILL RUNNING ??
843 003130  001405                          BEQ     2$              ; BR IF SO.
844 003132  005367  175140                  DEC     FCD1            ; NO, WE'RE DEAD -- DISPLAY...
845 003136  001061                          BNE     4$              ;...THE MESSAGE FOR A WHILE...
846 003140  000167  177320                  JMP     DONE2           ;...AND SIGNAL ALL DONE.
847
848 003144  016767  007026  007000  2$:     MOV     RB,ORB          ; OK, COPY THE ORIGINS...
849 003152  016767  007022  006774          MOV     RB+2,ORB+2      ;...SO WE CAN ERASE.
850 003160  016767  007024  006776          MOV     GB,OGB
851 003166  016767  007020  006772          MOV     GB+2,OGB+2
852 003174  016701  006776                  MOV     RB,R1           ;NOW LET'S SEE IF WE'VE...
853 003200  166701  007004                  SUB     GB,R1           ;...HAD A CRASH.
854 003204  100001                          BPL     .+4             ;IF DIFF NEG...
855 003206  005401                          NEG     R1              ;...MAKE POSITIVE.
856 003210  020127  000034                  CMP     R1,#34          ;X DIFF < 34 ??
857 003214  101025                          BHI     3$              ; NO, WE'RE OK !!!
858 003216  016701  006756                  MOV     RB+2,R1         ;YES, CRASH POSSIBLE...
859 003222  166701  006764                  SUB     GB+2,R1         ;...COMPARE Y ORIGINS.
860 003226  100001                          BPL     .+4             ;IF NEG DIFF...
861 003230  005401                          NEG     R1              ;...MAKE POSITIVE.
862 003232  020127  000034                  CMP     R1,#34          ;Y DIFF < 34 ??
863 003236  101014                          BHI     3$              ; BR IF NOT, WE'RE STILL OK.
864
865 003240  012767  051103  006754          MOV     #"CR,B3         ; OTHERWISE...
866 003246  012767  051501  006750          MOV     #"AS,B3+2       ;...SAY "CRASH" !!!!
867 003254  012767  020110  006744          MOV     #"H ,B3+4
868 003262  004767  000416                  JSR     PC,DING
869 003266  000402                          SKP2
870 003270  004767  000252          3$:     JSR     PC,MOVE         ; MOVE THE BALLS AROUND.
871 003274  005367  175012                  DEC     I1              ; DON'T RUN FOREVER,                    <<VER B>>
872 003300  001401                          BEQ     5$              ; GET OUT EVENTUALLY!                   <<VER B>>
873 003302  000677                  4$:     BR      CNTNU
874 003304  000167  177154          5$:     JMP     DONE2
875
876                                 ;
877                                 ; RANDOM NUMBER GETTER.
878                                 ;
879                                 ;
880 003310                          RAND:
    003310  104417  000000'                  RAND$,BEGIN
881 003314  016701  174536                  MOV     RANNUM,R1
882 003320  042701  176000                  BIC     #^C1777,R1
```



```
883 003324  020167  000026          1$:     CMP     R1,LOLIM
884 003330  103003                          BHIS    2$
885 003332  066701  000020                  ADD     LOLIM,R1
886 003336  000772                          BR      1$
887 003340  020167  000014          2$:     CMP     R1,HILIM
888 003344  101403                          BLOS    3$
889 003346  166701  000006                  SUB     HILIM,R1
890 003352  000772                          BR      2$
891 003354  000207                  3$:     RTS     PC
892 003356  000000                  LOLIM:  0
893 003360  000000                  HILIM:  0
```

```
                                ;NON EXISTANCE MEMORY TRAP HANDLER
                                ;
003362                          TRP1:
003362  016737  174754  000004          MOV     TEMPC,@#MEMERR          ;RESTORE NON EXIST MEM ADR
003370  016737  174744  000006          MOV     TEMPS,@#MEMERR+2        ;RESTORE PS
003376  022626                          CMP     (SP)+,(SP)+             ;ADJUST STACK
003400  000167  175760                  JMP     SETUP
003404                          TRP2:                                   ;!!!!TRP2 ADDED AT <<VER B>>!!!!
003404  016737  174732  000004          MOV     TEMPC,@#MEMERR          ;RESTORE NON EXIST MEM ADR
003412  016737  174722  000006          MOV     TEMPS,@#MEMERR+2        ;RESTORE PS
003420  022626                          CMP     (SP)+,(SP)+             ;ADJUST STACK
003422  016700  174652                  MOV     ADDRX,R0                ;GET CSR
003426  010067  174446                  MOV     R0,CSRA                 ;PUT INTO CSRA
003432  012767  000010  174446          MOV     #10,ERRTYP              ;LOAD ERROR CODE
003440  104405  000000' 000000          HRDERS,BEGIN,NULL       ;NXM TIMOUT,MODULE DROPPED
003446  104401  000000' 003466'          MSGS,BEGIN,2$   ;ASCII MESSAGE CALL
003454  046767  174624  174634          BIC     DEVON,T3                ;REMOVE FROM LIST
003462  000167  176776                  JMP     DONE2                   ;NEXT..!
                                ;
                                ;
003466      045     040     040 2$:     .ASCIZ  '%  VSCB0 -- NXM TIMEOUT,MODULE DROPPED%'
003471      126     123     103
003474      102     060     040
003477      055     055     040
003502      116     130     115
003505      040     124     111
003510      115     105     117
003513      125     124     054
003516      115     117     104
003521      125     114     105
003524      040     104     122
003527      117     120     120
003532      105     104     045
003535      000
                                .EVEN
                                ;
                                ;
                                ;
                                ; ROUTINE TO MOVE THE BALLS AROUND ON THE SCREEN.
                                ;
003536  000000  000000          DX1:    0,0                     ;DELTAS FOR BALL 1...
003542  000000  000000          DX2:    0,0                     ;...AND 2.
                                ;
003546  012701  012176'          MOVE:   MOV     #RB,R1          ;RED BALL ORIGIN...
003552  012702  003536'                  MOV     #DX1,R2         ;...AND DELTA POINTERS.
003556  004767  000010                  JSR     PC,1$
003562  012701  012210'                  MOV     #GR,R1          ; GREEN BALL ORIGIN...
003566  012702  003542'                  MOV     #DX2,R2         ;...AND DELTA POINTERS.
                                ;
003572  061211                  1$:     ADD     (R2),(R1)       ;OLD X + DELTA X.
003574  021127  000024                  CMP     (R1),#24        ;HIT LEFT WALL ??
003600  101005                          BHI     2$              ;NO
003602  012711  000024                  MOV     #24,(R1)        ;YES, SET X AT MIN.....
003606  005412                          NEG     (R2)            ;...AND REVERSE DELTA X.
```



```
003610  004767  000070                  JSR     PC,DING         ;"DING"
                                ;
003614  021127  001752          2$:     CMP     (R1),#MAXX-24   ;HIT RIGHT WALL ??
003620  103405                          BLO     3$              ;NO
003622  012711  001752                  MOV     #MAXX-24,(R1)   ;YES, SET X AT MAX...
003626  005412                          NEG     (R2)            ;...AND REVERSE DELTA X.
003630  004767  000050                  JSR     PC,DING
                                ;
003634  005721                  3$:     TST     (R1)+           ;BUMP POINTERS FOR Y'S.
003636  005722                          TST     (R2)+
003640  061211                          ADD     (R2),(R1)       ;OLD Y + DELTA Y.
003642  021127  000024                  CMP     (R1),#24        ;HIT FLOOR ??
003646  101005                          BHI     4$              ;NO
003650  012711  000024                  MOV     #24,(R1)        ;YES, SET Y AT MIN...
003654  005412                          NEG     (R2)            ;...AND REVERSE DELTA Y.
003656  004767  000022                  JSR     PC,DING
                                ;
003662  021127  001652          4$:     CMP     (R1),#MAXY60-24 ;HIT CEILING ??
003666  103405                          BLO     5$              ;NO
003670  012711  001652                  MOV     #MAXY60-24,(R1) ;YES, SET Y AT MAX...
003674  005412                          NEG     (R2)            ;...AND REVERSE DELTA Y.
003676  004767  000002                  JSR     PC,DING
                                ;
003702  000207                  5$:     RTS     PC              ;EXIT.
                                ;
003704  032767  000004  174104  DING:   BIT     #BIT2,SR1       ;IF BIT2=1 BELL OFF
003712  001003                          BNE     1$              ; EXIT IF "BELL" DISABLED.
003714  104401  000000' 003724'          MSGS,BEGIN,BELL ;ASCII MESSAGE CALL
003722  000207                  1$:     RTS     PC
003724      007     000         BELL:   .ASCIZ  <7>
```

```
                                        .SBTTL  *** PART 1 -- DISPLAY CODE
                                        ;
                                        ; SMORGASBORD, ONE FRAME THAT HAS A BIT OF EVERYTHING.
                                        ; FIRST, A DEC LOGO (SORT OF), THEN VIOLATE THE EDGES
                                        ; AT LOWER LEFT AND UPPER RIGHT.
                                        ;
          100000                        DSUB = CHAR             ; USE CHAR MODE TO CALL SUBROUTINES.
                                                                ; *** REMEMBER THAT THEY ***
                                                                ; *** CANNOT BE NESTED   ***
                                                                ; CHAR CODES 1 - 11, AND 16 - 77
                                                                ; ARE AVAILABLE AS CALL ARGUMENTS.
                                                                ; APPROPRIATE DISPATCH POINTERS MUST
                                                                ; BE SET IN TABLE "ACAT".

003726  176074  176474           DBEGIN: RDWRT!CH0, RDWRT!CH1
003732  177074  177474                   RDWRT!CH2, RDWRT!CH3
003736  152000  012320'                  SETCB, ACAT             ; SET CHAR BASE POINTER.
003742  170140  173000                   CLRMEM, STOPN           ; CLEAR SCREEN RAM AND STOP

003746  114000  000540  000100   LOGO:   APNT, 540, 100
003754  103774                   CHAR!ALL
003756     104     040     111           .ASCIZ  /D I G I T A L/
003761     040     107     040
003764     111     040     124
003767     040     101     040
003772     114     000

003774  114000  000140  000140           APNT,140,140            ;EXCEED EDGES AT LOWER LEFT...
004002  110000                           LVEC
004004  060200  000000                   .WORD   $I+$X, $Y
004010  040000  020200                   .WORD   $I+$X, $Y
004014  040200  000000                   .WORD   $I+$X, $Y
004020  040000  000200                   .WORD   $I+$X, $Y
004024  114000  001640  001540           APNT,1640,1540          ;...AND AT UPPER RIGHT.
004032  110000                           LVEC
004034  040200  000000                   .WORD   $I+$X, $Y
004040  040000  000300                   .WORD   $I+$X, $Y
004044  060200  000000                   .WORD   $I+$X, $Y
004050  040000  020300                   .WORD   $I+$X, $Y
004054  173401                           STOPI+1                 ; END OF FRAME 1.
```



```
                                        ;
                                        ; NOW LONG, AND SHORT VECTORS, AND SOME RELATIVE POINTS.
                                        ;
004056  114000  000500  000002   OCTGNS: APNT,500,2
004064  113774                           LVEC!ALL                ; 1 LARGE OCTAGON...
004066  040576  000000                   .WORD   $I+$X, $Y
004072  040436  000436                   .WORD   $I+$X, $Y
004076  040000  000576                   .WORD   $I+$X, $Y
004102  060436  000436                   .WORD   $I+$X, $Y
004106  060576  000000                   .WORD   $I+$X, $Y
004112  060436  020436                   .WORD   $I+$X, $Y
004116  040000  020576                   .WORD   $I+$X, $Y
004122  040436  020436                   .WORD   $I+$X, $Y

004126  114000  000160  000620           APNT,160,620            ;...AND 2 SMALL ONES.
004134  100000  000001                   DSUB, 1                 ; CALL OCTAGON SUBROUTINE.

004140  114000  001520  000620           APNT,1520,620
004146  100000  000001                   DSUB, 1

004152  173402                           STOPI+2                 ; END OF FRAME 2.

004154                           SUB.1:                  ; SUBROUTINE FOR SMALL OCTAGON...
004154  107774                           SVEC!ALL                ;...CALLED VIA CHAR CODE 1.
004156  057400                           .WORD   $I+$X+$Y
004160  053456                           .WORD   $I+$X+$Y
004162  040076                           .WORD   $I+$X+$Y
004164  073456                           .WORD   $I+$X+$Y
004166  077400                           .WORD   $I+$X+$Y
004170  073556                           .WORD   $I+$X+$Y
004172  040176                           .WORD   $I+$X+$Y
004174  053556                           .WORD   $I+$X+$Y
004176  130000                           RPNT                    ;5 REL POINTS IN CENTER.
004200  004050                           .WORD   $I+$X+$Y
004202  044050                           .WORD   $I+$X+$Y
004204  044000                           .WORD   $I+$X+$Y
004206  064020                           .WORD   $I+$X+$Y
004210  064120                           .WORD   $I+$X+$Y
004212  044120                           .WORD   $I+$X+$Y
004214  165000                           DPOP
```

```
1048                                    ;
1049                                    ; NEXT, SHOW GRAPH/HISTO Y IN UPPER LEFT.
1050                                    ;
1051            001240                  YVAL = 1240
1052 004216  114000  000100  001240  GRY:    APNT,100,YVAL
1053 004224  113774                          LVEC!ALL                ;DRAW BASE LINE.
1054 004226  040340  000000                  I!340, 0
1055 004232  020344  000000                  MXY!344, 0
1056 004236  174104                          GXI!4                   ;X INCR = 4 FOR GRAPH.
1057 004240  127774                          GHY!ALL                 ; GRAPH Y MODE.
1058            000071                       .REPT   71
1059                                         0!YVAL
1060                                         YVAL=YVAL+4
1061                                         .ENDM
     004242  001240                         0!YVAL
     004244  001244                         0!YVAL
     004246  001250                         0!YVAL
     004250  001254                         0!YVAL
     004252  001260                         0!YVAL
     004254  001264                         0!YVAL
     004256  001270                         0!YVAL
     004260  001274                         0!YVAL
     004262  001300                         0!YVAL
     004264  001304                         0!YVAL
     004266  001310                         0!YVAL
     004270  001314                         0!YVAL
     004272  001320                         0!YVAL
     004274  001324                         0!YVAL
     004276  001330                         0!YVAL
     004300  001334                         0!YVAL
     004302  001340                         0!YVAL
     004304  001344                         0!YVAL
     004306  001350                         0!YVAL
     004310  001354                         0!YVAL
     004312  001360                         0!YVAL
     004314  001364                         0!YVAL
     004316  001370                         0!YVAL
     004320  001374                         0!YVAL
     004322  001400                         0!YVAL
     004324  001404                         0!YVAL
     004326  001410                         0!YVAL
     004330  001414                         0!YVAL
     004332  001420                         0!YVAL
     004334  001424                         0!YVAL
     004336  001430                         0!YVAL
     004340  001434                         0!YVAL
     004342  001440                         0!YVAL
     004344  001444                         0!YVAL
     004346  001450                         0!YVAL
     004350  001454                         0!YVAL
     004352  001460                         0!YVAL
     004354  001464                         0!YVAL
     004356  001470                         0!YVAL
     004360  001474                         0!YVAL
     004362  001500                         0!YVAL
     004364  001504                         0!YVAL
     004366  001510                         0!YVAL
```



```
     004370  001514                         0!YVAL
     004372  001520                         0!YVAL
     004374  001524                         0!YVAL
     004376  001530                         0!YVAL
     004400  001534                         0!YVAL
     004402  001540                         0!YVAL
     004404  001544                         0!YVAL
     004406  001550                         0!YVAL
     004410  001554                         0!YVAL
     004412  001560                         0!YVAL
     004414  001564                         0!YVAL
     004416  001570                         0!YVAL
     004420  001574                         0!YVAL
     004422  001600                         0!YVAL
1062
1063            001240                  YVAL = 1240
1064 004424  150000  001240          HSY:    SETHB,YVAL              ; SET HISTO BASE LINE.
1065 004430  114000  000130  001240          APNT,130,YVAL
1066 004436  174110                          GXI!10                  ;X INCR = 10 FOR HISTO.
1067 004440  127774                          GHY!ALL
1068            000031                       .REPT   31
1069                                         HST!YVAL
1070                                         YVAL=YVAL+10
1071                                         .ENDR
     004442  041240                         HST!YVAL
     004444  041250                         HST!YVAL
     004446  041260                         HST!YVAL
     004450  041270                         HST!YVAL
     004452  041300                         HST!YVAL
     004454  041310                         HST!YVAL
     004456  041320                         HST!YVAL
     004460  041330                         HST!YVAL
     004462  041340                         HST!YVAL
     004464  041350                         HST!YVAL
     004466  041360                         HST!YVAL
     004470  041370                         HST!YVAL
     004472  041400                         HST!YVAL
     004474  041410                         HST!YVAL
     004476  041420                         HST!YVAL
     004500  041430                         HST!YVAL
     004502  041440                         HST!YVAL
     004504  041450                         HST!YVAL
     004506  041460                         HST!YVAL
     004510  041470                         HST!YVAL
     004512  041500                         HST!YVAL
     004514  041510                         HST!YVAL
     004516  041520                         HST!YVAL
     004520  041530                         HST!YVAL
     004522  041540                         HST!YVAL
1072 004524  173403                         STOPI+3                 ; END OF FRAME 3.
```

```
                                        ;
                                        ; NEXT, GRAPH/HISTO X IN LOWER RIGHT.
                                        ;
                 001340                 XVAL = 1340
004526  114000  001340  000100  GRX:    APNT,XVAL,100
004534  113774                          LVEC!ALL                ;DRAW BASE LINE.
004536  040000  000340                   I!0, 340
004542  000000  020344                   0, MXY!344
004546  174104                          GYI!4                   ;Y INCR = 4 FOR GRAPH.
004550  123774                          GHX!ALL
                 000071                 .REPT   71
                                        0!XVAL
                                        XVAL=XVAL+4
                                        .ENDR
     004552  001340                     0!XVAL
     004554  001344                     0!XVAL
     004556  001350                     0!XVAL
     004560  001354                     0!XVAL
     004562  001360                     0!XVAL
     004564  001364                     0!XVAL
     004566  001370                     0!XVAL
     004570  001374                     0!XVAL
     004572  001400                     0!XVAL
     004574  001404                     0!XVAL
     004576  001410                     0!XVAL
     004600  001414                     0!XVAL
     004602  001420                     0!XVAL
     004604  001424                     0!XVAL
     004606  001430                     0!XVAL
     004610  001434                     0!XVAL
     004612  001440                     0!XVAL
     004614  001444                     0!XVAL
     004616  001450                     0!XVAL
     004620  001454                     0!XVAL
     004622  001460                     0!XVAL
     004624  001464                     0!XVAL
     004626  001470                     0!XVAL
     004630  001474                     0!XVAL
     004632  001500                     0!XVAL
     004634  001504                     0!XVAL
     004636  001510                     0!XVAL
     004640  001514                     0!XVAL
     004642  001520                     0!XVAL
     004644  001524                     0!XVAL
     004646  001530                     0!XVAL
     004650  001534                     0!XVAL
     004652  001540                     0!XVAL
     004654  001544                     0!XVAL
     004656  001550                     0!XVAL
     004660  001554                     0!XVAL
     004662  001560                     0!XVAL
     004664  001564                     0!XVAL
     004666  001570                     0!XVAL
     004670  001574                     0!XVAL
     004672  001600                     0!XVAL
     004674  001604                     0!XVAL
     004676  001610                     0!XVAL
```



```
     004700  001614                     0!XVAL
     004702  001620                     0!XVAL
     004704  001624                     0!XVAL
     004706  001630                     0!XVAL
     004710  001634                     0!XVAL
     004712  001640                     0!XVAL
     004714  001644                     0!XVAL
     004716  001650                     0!XVAL
     004720  001654                     0!XVAL
     004722  001660                     0!XVAL
     004724  001664                     0!XVAL
     004726  001670                     0!XVAL
     004730  001674                     0!XVAL
     004732  001700                     0!XVAL

                 001340                 XVAL = 1340
004734  150000  001340          HSX:    SETHB,XVAL              ; SET HISTO BASE LINE.
004740  114000  001340  000130          APNT,XVAL,130
004746  174110                          GYI!10                  ;Y INCR = 10 FOR HISTO.
004750  123774                          GHX!ALL
                 000031                 .REPT   31
                                        HST!XVAL
                                        XVAL=XVAL+10
                                        .ENDR
     004752  041340                     HST!XVAL
     004754  041350                     HST!XVAL
     004756  041360                     HST!XVAL
     004760  041370                     HST!XVAL
     004762  041400                     HST!XVAL
     004764  041410                     HST!XVAL
     004766  041420                     HST!XVAL
     004770  041430                     HST!XVAL
     004772  041440                     HST!XVAL
     004774  041450                     HST!XVAL
     004776  041460                     HST!XVAL
     005000  041470                     HST!XVAL
     005002  041500                     HST!XVAL
     005004  041510                     HST!XVAL
     005006  041520                     HST!XVAL
     005010  041530                     HST!XVAL
     005012  041540                     HST!XVAL
     005014  041550                     HST!XVAL
     005016  041560                     HST!XVAL
     005020  041570                     HST!XVAL
     005022  041600                     HST!XVAL
     005024  041610                     HST!XVAL
     005026  041620                     HST!XVAL
     005030  041630                     HST!XVAL
     005032  041640                     HST!XVAL
005034  173404                          STOPI+4                 ; END OF FRAME 4.
```

```
                                        ;
                                        ; NEXT SHOW BIT MAP MODE 1 AT BOTTOM CENTER.
                                        ; AND BIT MAP MODE 0 AT TOP CENTER.
                                        ;
                                        ; DON'T FORGET TO FILL THE BUFFER FIRST.
                                        ;
005036  114000  000400  000240  B14:    APNT,400,240            ; 4 ON 4 OFF FOR 340 PIXELS.
005044  136340                          BM14!340
005046  005132'                         DBUFR
005050  164000                          DNOP

005052  114000  000300  000340  B18:    APNT,300,340            ; 2 ON 2 OFF FOR 400 PIXELS.
005060  137400                          BM18!400
005062  005132'                         DBUFR
005064  164000                          DNOP

005066  114000  000540  001340  B04:    APNT, 540, 1340         ; 4 BIT X 32 SQUARE...
005074  134010                          BM04!M32!EX2            ;...EXPANDED X 2.
005076  005132'                         DBUFR
005100  164000                          DNOP

005102  114000  001040  001340  B08:    APNT, 1040, 1340        ; 8 BIT X 32 SQUARE...
005110  135010                          BM08!M32!EX2            ;...EXPANDED X 2.
005112  005132'                         DBUFR
005114  173405                          STOP1+5                 ; END OF FRAME 5.

                                        ;
                                        ; FINALLY, AT DEAD-CENTER:
                                        ; VERTICAL COLOR BARS -- NUMBER OF COLORS/SHADES DEPENDANT
                                        ; UPON IMAGE MEMORY SIZE AND PRESENCE (OR ABSENCE) OF THE LUT.
                                        ; IF ALL 256. GREY-SCALE LEVELS ARE ACTIVE, WE'LL HAVE ONLY
                                        ; 1 VECTOR PER LEVEL !!!!!
                                        ; BUFFER REQUIRES <256.*5>+2 WORDS.
                                        ;

005116  114000  000376  000476  BARS:   APNT, MAPX-400, MAPY60-240
005124  100000  000002                  DSUB, 2                 ; EXECUTE CODE IN BUFFER.
005130  173406                          STOP1+6                 ; END OF FRAME 6.
                                        ;
                                        ; DISPLAY CODE BUFFER.
                                        ; SHARED BY THE BIT MAP AND BARS ROUTINES.
                                        ;
005132                                  SUB.2:                  ; SUBROUTINE CALL FOR "BARS".
005132                                  DBUFR:  .BLKW   <256.*5>+2
```



```
                                        .SBTTL  *** PART 2 -- DISPLAY CODE
                                        ;
                                        ; DISPLAY FILES FOR BOUNCING BALLS.
                                        ; IMAGE MEMORIES ARE IN R/W MODE INITIALLY.
                                        ; IF R/W MODE SELECTED, USE B1A AS ENTRY.
                                        ; OTHERWISE, RECONFIGURE IM'S AT B1, AND USE B2 AS THE ENTRY.
                                        ;
012136  176050  176434          B1:     READ!CH0, WRT!CH1       ; DYNAMIC MODE, RESET IM'S.
012142  177050  177434                  READ!CH2, WRT!CH3
012146  173000                          STOPN

012150  116000                  B1A:    APNT!NONE               ; READ/WRITE MODE...
012152  000000  000000          ORB:    0,0                     ;...ERASE OLD IMAGE...
012156  100000  000003                  DSUB, 3
012162  114000                          APNT
012164  000000  000000          OGB:    0,0
012170  100000  000003                  DSUB, 3                 ;...AND FALL THRU.

012174  116500                  B2:     APNT!RED                ; DYNAMIC MODE...
012176  000000  000000          RB:     0,0                     ;...REDRAW NEW IMAGE.
012202  100000  000003                  DSUB, 3
012206  117140                          APNT!GRN
012210  000000  000000          GB:     0,0
012214  100000  000003                  DSUB, 3
012220  103774                          CHAR!WHITE
012222     040     040     040  B3:     .ASCII  /       /         ; "CRASH" IF THAT'S THE CASE.
012225     040     040     040
012230  117774  000000  000000          APNT!WHITE, 0, 0        ; FINALLY, REFRESH THE FRAME.
012236  110000                          LVEC
012240  041776  000000                  1+MAXX, 0
012244  040000  001676                  I+0, MAXY60
012250  061776  000000                  I+MXY+MAXX, 0
012254  040000  021676                  I+0, MXY+MAXY60

012260  164002                  B4:     SYNC+1                  ; "SWITCH!CURMEM" IF DYNAMIC.
012262  164000  164000                  DNOP, DNOP
012266  173400                          STOP1
                                        ;
                                        ; OCTAGON SUBROUTINE.
                                        ;
012270                                  SUB.3:                  ; CALL VIA CHAR CODE 3.
012270  104000                                  SVEC            ;DRAW CCW FROM LO LEFT CORNER.
012272  022124                                  .WORD   $I+$X+$Y
012274  044000                                  .WORD   $I+$X+$Y
012276  043014                                  .WORD   $I+$X+$Y
012300  040020                                  .WORD   $I+$X+$Y
012302  063014                                  .WORD   $I+$X+$Y
012304  064000                                  .WORD   $I+$X+$Y
012306  063114                                  .WORD   $I+$X+$Y
012310  040120                                  .WORD   $I+$X+$Y
012312  043114                                  .WORD   $I+$X+$Y
012314  002024                                  .WORD   $I+$X+$Y
012316  165000                                  DPOP
```

```
                                        .SBTTL  *** ASCII CHARACTER ADDRESS TABLE AND SUBPIX
                                        ;
                                        ;THE FOLLOWING 128. WORDS CONTAIN THE CHARACTER SUBPIX
                                        ;STARTING ADDRESSES.  0-37 AND 140-177 ARE NULL FOR NOW.
                                        ;       (EXCEPT FOR 12 AND 15)
                                        ;
012320  012720'                 ACAT:   ..NULL          ;0-37 ARE NULL (NLISTED).

        012420'                         $SVPC=.         ;SAVE PC, BACK UP AND...
        012344'                         .=ACAT+<12*2>
012344  012722'                         ..LF            ;...INSERT LINE FEED...
        012352'                         .=ACAT+<15*2>
012352  012732'                         ..CR            ;...AND CARRIAGE RETURN ADDRESSES.
        012420'                         .=$SVPC         ;RESTORE PC

012420  012746'                         ..SPC           ;SPACE
012422  012754'                         ..EXC           ;EXCLAM
012424  012772'                         ..QUO           ;DOUBLE QUOTE.
012426  013010'                         ..NUM           ;NUMBER SIGN
012430  013036'                         ..DOL           ;DOLLAR
012432  013070'                         ..PCT           ;PER CENT
012434  013130'                         ..AND           ;AMPERSAND
012436  013170'                         ..QUO1          ;SINGLE QUOTE
012440  013202'                         ..LPAR          ;LFT PAREN
012442  013220'                         ..RPAR          ;RT PAREN
012444  013236'                         ..AST           ;ASTERISK
012446  013260'                         ..PLS           ;PLUS
012450  013276'                         ..CMA           ;COMMA
012452  013310'                         ..MNS           ;MINUS
012454  013322'                         ..DOT           ;PERIOD
012456  013332'                         ..SLSH          ;SLASH
012460  013346'                         ..00
012462  013400'                         ..11
012464  013420'                         ..22
012466  013452'                         ..33
012470  013512'                         ..44
012472  013532'                         ..55
012474  013562'                         ..66
012476  013614'                         ..77
012500  013634'                         ..88
012502  013704'                         ..99
012504  013734'                         ..COLN          ;COLON
012506  013752'                         ..SEMI          ;SEMI COLON
012510  013770'                         ..LANG          ;LFT ANGLE
012512  014004'                         ..EQ            ;EQUALS
012514  014022'                         ..RANG          ;RT ANGLE
012516  014034'                         ..QUES          ;QUESTION
012520  014062'                         ..ATS           ;AT SIGN
012522  014116'                         ..AA
012524  014142'                         ..BB
012526  014176'                         ..CC
012530  014224'                         ..DD
012532  014246'                         ..EE
012534  014254'                         ..FF
012536  014272'                         ..GG
012540  014310'                         ..HH
012542  014330'                         ..II
```



```
012544  014352'                         ..JJ
012546  014372'                         ..KK
012550  014410'                         ..LL
012552  014424'                         ..MM
012554  014442'                         ..NN
012556  014462'                         ..OO
012560  014512'                         ..PP
012562  014534'                         ..QQ
012564  014550'                         ..RR
012566  014564'                         ..SS
012570  014622'                         ..TT
012572  014640'                         ..UU
012574  014662'                         ..VV
012576  014702'                         ..WW
012600  014722'                         ..XX
012602  014746'                         ..YY
012604  014772'                         ..ZZ
012606  015014'                         ..LBRK          ;LFT BRACKET
012610  015032'                         ..BSLH          ;BACK SLASH
012612  015050'                         ..RBRK          ;RT BRACKET
012614  015066'                         ..CRT           ;CARROT
012616  015102'                         ..USC           ;UNDERSCORE
012620  012720'                         ..NULL          ;140-177 ARE NULL (NLISTED).
```

```
                                ; THESE ARE THE CHAR SUBPIX ROUTINES.
                                ; EACH IS BUILT ON A 5 X 7 DOT MATRIX (ALA VT05).
                                ; <CR> WILL INVOKE A "STOP".  CPU MUST CALCULATE THE START
                                ; CO-ORDS FOR NEXT LINE, STUFF IN "..CRX AND ..CRX+2" AND
                                ; RESUME, TO XCT PSUEDO-CRLF.
                                ;
                                ; THE FOLLOWING MACRO (SXY4) APPLIES A 4X SCALE FACTOR AT
                                ; ASSEMBLY TIME TO YIELD A CHAR SIZE OF 30 X 40 (OCTAL)
                                ; WHICH PRODUCES A MEDIUM SIZE CHAR FONT.
                                ;
                                        .MACRO  SXY4 A,B,C
                                        SXY     A,B*4,C*4
                                        .ENDM   SXY4

1303 012720  165000             ..NULL: DPOP                    ;ALL UNDEFINED CHARS EXIT HERE.

                                ;****** CHAR SET IS .NLISTED ***********
```



```
                                .SBTTL  *** 256 LEVEL LUMINANCE (GREY-SCALE) TABLE
                                ;
                                ; THIS IA A TABLE OF 256 (8 BITS) LUMINANCE VALUES
                                ; WHICH SHOULD PRODUCE A LINEAR 12 BIT GREY-SCALE.
                                ;
                                ; ON THE GREY-SCALE, 36 = BLUE, 120 = RED, AND 230 = GREEN.
                                ;
015112                          LUMTBL:
015112 000000 000400 001000     0,      400,    1000,   1
015120 000001
015122 001400 000401 002000     1400,   401,    2000,   1001
015130 001001
015132 002400 000002 003000     2400,   2,      3000,   402
015140 000402
015142 003400 002401 004000     3400,   2401,   4000,   3
015150 000003
015152 003001 002020 005000     3001,   2020,   5000,   2420
015160 002420
015162 000004 005400 003002     4,      5400,   3002,   6000
015170 006000
015172 001040 006400 002403     1040,   6400,   2403,   5
015200 000005
015202 003003 002040 007400     3003,   2040,   7400,   5020    ; LEVEL 37
015210 005020
015212 002404 000060 000006     2404,   60,     6,      1441
015220 001441
015222 003004 002023 001042     3004,   2023,   1042,   2405
015230 002405
015232 000007 004440 002024     7,      4440,   2024,   1061
015240 001061
015242 000100 000010 000462     100,    10,     462,    2025
015250 002025
015252 001044 002407 000011     1044,   2407,   11,     501
015260 000501
015262 007440 001063 000120     7440,   1063,   120,    12
015270 000012
015272 000502 003500 001064     502,    3500,   1064,   2411
015300 002411
015302 005407 000503 002030     5407,   503,    2030,   1065    ; LEVEL 77
015310 001065
015312 000140 000014 000504     140,    14,     504,    2031
015320 002031
015322 001066 002413 000015     1066,   2413,   15,     505
015330 000505
015332 007444 002540 000160     7444,   2540,   160,    16
015340 000016
015342 003014 003504 002541     3014,   3504,   2541,   125
015350 000125
015352 000017 002160 002034     17,     2160,   2034,   200
015360 000200
015362 000126 003142 002161     126,    3142,   2161,   3506
015370 003506
015372 001072 001600 007066     1072,   1600,   7066,   2162
015400 002162
015402 002036 000220 001601     2036,   220,    1601,   3126    ; LEVEL 137
015410 003126
015412 002163 005161 002563     2163,   5161,   2563,   131
```

```
        015420  000131
   1912 015422  004600  000513  005162  4600,   513,    5162,   240
        015430  000240
   1913 015432  000132  003146  002165  132,    3146,   2165,   1222
        015440  001222
   1914 015442  004220  004074  004602  4220,   4074,   4602,   2166
        015450  002166
   1915 015452  003513  000260  004075  3513,   260,    4075,   3132
        015460  003132
   1916 015462  003640  001152  002477  3640,   1152,   2477,   3223
        015470  003223
   1917 015472  000607  003567  000300  607,    3567,   300,    210
        015500  000210
   1918 015502  005476  002243  005241  5476,   2243,   5241,   1154    ; LEVEL 177
        015510  001154
   1919 015512  004152  004660  003153  4152,   4660,   3153,   2172
        015520  002172
   1920 015522  003517  000320  004117  3517,   320,    4117,   646
        015530  000646
   1921 015532  002173  001230  002555  2173,   1230,   2555,   157
        015540  000157
   1922 015542  000647  007624  000340  647,    7624,   340,    2574
        015550  002574
   1923 015552  005554  002321  002213  5554,   2321,   2213,   2703
        015560  002703
   1924 015562  000267  003265  004612  267,    3265,   4612,   3647
        015570  003647
   1925 015572  000360  001741  003266  360,    1741,   3266,   4631
        015600  004631
   1926 015602  002177  005213  005703  2177,   5213,   5703,   1634    ; LEVEL 237
        015610  001634
   1927 015612  000671  003651  005176  671,    3651,   5176,   272
        015620  000272
   1928 015622  001617  006266  001362  1617,   6266,   1362,   4360
        015630  004360
   1929 015632  000273  007232  000655  273,    7232,   655,    3653
        015640  003653
   1930 015642  001237  004235  004743  1237,   4235,   4743,   1256
        015650  001256
   1931 015652  007615  001312  005237  7615,   1312,   5237,   3345
        015660  003345
   1932 015662  000731  007616  002712  731,    7616,   2712,   1747
        015670  001747
   1933 015672  006235  002331  003656  6235,   2331,   3656,   4364
        015700  004364
   1934 015702  000277  003275  002332  277,    3275,   2332,   5330    ; LEVEL 277
        015710  005330
   1935 015712  002714  005712  003276  2714,   5712,   3276,   7745
        015720  007745
   1936 015722  001370  004366  007364  1370,   4366,   7364,   3277
        015730  003277
   1937 015732  002334  002371  004367  2334,   2371,   4367,   5714
        015740  005714
   1938 015742  004751  006276  001372  4751,   6276,   1372,   2717
        015750  002717
   1939 015752  005715  004752  006277  5715,   4752,   6277,   1373
        015760  001373
```



```
   1940 015762  004371  007367  004753  4371,   7367,   4753,   2337
        015770  002337
   1941 015772  001374  004372  005717  1374,   4372,   5717,   4754
        016000  004754
   1942 016002  007752  001375  004373  7752,   1375,   4373,   1757    ; LEVEL 337
        016010  001757
   1943 016012  004755  007753  005337  4755,   7753,   5337,   4374
        016020  004374
   1944 016022  007372  003357  006355  7372,   3357,   6355,   7754
        016030  007754
   1945 016032  005374  002776  005774  5374,   2776,   5774,   7337
        016040  007337
   1946 016042  006356  005357  004376  6356,   5357,   4376,   7374
        016050  007374
   1947 016052  004776  007756  006757  4776,   7756,   6757,   4377
        016060  004377
   1948 016062  007375  006376  007757  7375,   6376,   7757,   2777
        016070  002777
   1949 016072  006776  005777  006376  6776,   5777,   6376,   6377
        016100  006377
   1950 016102  007776  006777  007377  7776,   6777,   7377,   7777    ; LEVEL 377
        016110  007777
```

```
1952                                .SBTTL  *** PATCH AREA AND END
1953                                ;
1954                                ; PATCHES GO HERE.
1955                                ;
1956 016112                        PATCH:  .BLKW   40
1957
1958            016212'             LASTAD= .               ; END OF MODULE.
1959
1960                                .SBTTL  ****** TH - TH - TH - TH - THAT'S ALL FOLKS ! ******
1961            000001                      .END
```



```
ACAT    012320R  CLRMEM= 170140  GHY    = 124000  MOVE    003546R  RES2    000060R
ACSR    000102R  CNTNU   003102R  GHASCL  000626R  MSGN$ = 104403  RPNT  = 130000
ADDBAS= 000010   CONFIG  000056R  GRN    = 003140  MSGS$ = 104402  RSTRT   000112R
ADDR    000006R  CSRA    000100R  GRX     004526R  MSGS  = 104401  R256  = 000100
ADDRX   000300R  CUIM  = 146016   GRY     004216R  MSX   = 020000  R6    =%000006
ADDR22= 001000   CUIOFF= 146012   GWBUFS= 104414  MSY   = 000100  R7    =%000007
ALL   = 003774   CUIS  = 146013   GXI    = 174100  MXY   = 020000  SAVDPU  000452R
APNT  = 114000   CUOFF = 146040   GYI    = 174100  M128  = 000002  SBADR   000102R
ASB     000106R  CUON  = 146060   HAFX   = 000776  M256  = 000003  SDPC    000252R
ASTAT   000104R  CURD  = 146100   HAFY   = 000776  M32   = 000000  SDSR    000254R
AWAS    000110R  DATCK$= 104411   HAFY50= 000776  M64   = 000001  SDXR    000256R
BADPAD  000336R  DATER$= 104404   HAFY60= 000736  NEXT    001774R  SDYR    000260R
BARS    005116R  DBEGIN  003726R  HILIM   003360R  NONE  = 002000  SETCB = 152000
BEGIN   000000R  DBUFR   005132R  HRDCNT  000044R  NULL  = 000000  SETHB = 150000
BELL    003724R  DEVON   000304R  HRDER$= 104405  OCTGNS  004056R  SETMEM= 170100
BIT0  = 000001   DING    003704R  HRDPAS  000050R  OGB     012164R  SETRR = 000014
BIT1  = 000002   DIVCHK  000306R  HS1    = 040000  ONE     001042R  SETUP   001364R
BIT10 = 002000   DJMP  = 160000   HSX     004734R  OPEN  = 000000  SHADES  000674R
BIT11 = 004000   DLOOP   000330R  HSY     004424R  ORB     012152R  SIOFF = 171000
BIT12 = 010000   DNOP  = 164000   HUES    002354R  OTOA$ = 104420  SION  = 171400
BIT13 = 020000   DONE1   002454R  I      = 040000  PADDR   000322R  SKP1  = 000401
BIT14 = 040000   DONE2   002464R  ICONT   000036R  PART1   001632R  SKP2  = 000402
BIT15 = 100000   DPC     000222R  ICOUNT  000040R  PART2   002546R  SKP3  = 000403
BIT2  = 000004   DPOP  = 165000   IDNUM   000122R  PASCNT  000034R  SLDR    000266R
BIT3  = 000010   DRR     000224R  INIT    000030R  PASS    002536R  SLMR    000270R
BIT4  = 000020   DSR     000224R  INTR    000120R  PATCH   016112R  SLSR    000264R
BIT5  = 000040   DSTMT   000326R  JSMV    000242R  PIRQ$ = 000004  SOFCNT  000042R
BIT6  = 000100   DSUB  = 100000   JSSV    000246R  POPSP = 005726  SOFER$= 104406
BIT7  = 000200   DVID1   000014R  JS0    = 000000  POPSP2= 022626  SOFPAS  000046R
BIT8  = 000400   DX     = 000002  JS1    = 000400  PROTEC= 176000  SPUINT  000032R
BIT9  = 001000   DXR     000226R  JS2    = 001000  PRTY  = 000000  SPSIZ = 000040
BLU   = 002170   DX1     003536R  JS3    = 001400  PRTY0 = 000000  SR1     000016R
BLXIT   000756R  DX2     003542R  LASTAD= 016212R  PRTY1 = 000040  SR2     000020R
BM04  = 134000   DYI   = 000002   LDI    = 002000  PRTY2 = 000100  SR3     000022R
BM08  = 135000   DYNI  = 000004   LDR     000234R  PRTY3 = 000140  SR4     000024R
BM14  = 136000   DYR     000230R  LDUN   = 100000  PRTY4 = 000200  START   001000R
BM18  = 137000   EABITS  000324R  LMR     000236R  PRTY5 = 000240  STAT    000026R
BREAK$= 104407   ENDITS= 104413   LOGO    003746R  PRTY6 = 000300  STOP  = 172000
BR1     000012R  END$  = 104410   LOLIM   003356R  PRTY7 = 000340  STOPI = 173400
BR2     000013R  ERRTYP  000106R  LREAD = 010000  PS    = 177776  STOPN = 173000
BTOD$ = 104421   EXITS = 104400   LSET  = 010000  PSW   = 177776  STOP1   002022R
B04     005066R  EXSM2 = 000050   LSR     000232R  PUSH  = 005746  STOP2   003114R
B08     005102R  EXSM4 = 000060   LUMTBL  015112R  PUSH2 = 024646  STPV    000240R
B1      012136R  EX2   = 000010   LUIFLG  000262R  RAND    003310R  SUB.1   004154R
B1A     012150R  EX4   = 000020   LUTIM   001256R  RAND$ = 104417  SUB.2   005132R
B14     005036R  FCD     000274R  LUTV    000250R  RANNUM  000054R  SUB.3   012270R
B18     005052R  FCD1    000276R  LVEC  = 110000  RB      012176R  SVEC  = 104000
B2      012174R  FIDX    000272R  L0    = 002000  RDWRT = 176074  SVR0    000062R
B3      012222R  FR2     002076R  MAP22$= 104416  RDWRT1= 176076  SVR1    000064R
B4      012260R  FR3     002106R  MATCH   000344R  READ  = 176050  SVR2    000066R
CANT    001564R  FR4     002116R  MAXX  = 001776  RED   = 002500  SVR3    000070R
CDATA$= 104412   FR5     002130R  MAXY  = 001776  RELOC   000332R  SVR4    000072R
CHAR  = 100000   FR6     002224R  MAXY50= 001776  RELOK   000334R  SVR5    000074R
CH0   = 000000   GB      012210R  MAXY60= 001676  RESM    002012R  SVR6    000076R
CH1   = 000400   GCOFF = 000100   MEMERR= 000004  RESTRA  001064R  SWE   = 000010
CH2   = 001000   GETPA$= 104415   MODNAM  000000R  RESTRT  001026R  SWITCH  000344R
CH3   = 001400   GHX   = 120000   MODSP   000222R  RES1    000056R  SWTCH = 170200
```

SEQ 0043

```
SYNC   = 164001          WDTO    000114R         ..CR    012732R         ..MM    014424R         ..SS    014564R
SYSCNT  000052R          WHITE = 003774          ..CRT   015066R         ..MNS   013310R         ..TT    014622R
TEMPC   000342R          WR1   = 176034          ..CRX   012740R         ..NN    014442R         ..USC   015102R
TEMPS   000340R          WR11  = 176036          ..DD    014224R         ..NULL  012720R         ..UU    014640R
TIMOUT  000366R          XFLAG   000005R         ..DUL   013036R         ..NUM   013010R         ..VV    014662R
TOTV    000244R          XVAL  = 001650          ..DOT   013322R         ..OO    014462R         ..WW    014702R
TRPOFF= 000022           YVAL  = 001550          ..EE    014246R         ..PCT   013070R         ..XX    014722R
TRP1    003362R          $1    = 000000          ..EQ    014004R         ..PLS   013260R         ..YY    014746R
TRP2    003404R          $SVPC = 012420R         ..EXC   012754R         ..PP    014512R         ..ZZ    014772R
T0      000310R          $X    = 002000          ..FF    014254R         ..QQ    014534R         ..00    013346R
T1      000312R          $Y    = 000000          ..GG    014272R         ..QUES  014034R         ..11    013400R
T2      000314R          ..AA    014116R         ..HH    014310R         ..QUO   012772R         ..22    013420R
T3      000316R          ..AND   013130R         ..II    014330R         ..QUO1  013170R         ..33    013452R
UNXLI   000512R          ..AST   013236R         ..JJ    014352R         ..RANG  014022R         ..44    013512R
VADDR   000320R          ..ATS   014062R         ..KK    014372R         ..RBRK  015050R         ..55    013532R
VECBAS= 000020           ..BB    014142R         ..LANG  013770R         ..RPAR  013220R         ..66    013562R
VECTOR  000010R          ..BSLH  015032R         ..LBRK  015014R         ..RR    014550R         ..77    013614R
VECTXX  000302R          ..CC    014176R         ..LF    012722R         ..SEMI  013752R         ..88    013634R
WASADR  000104R          ..CMA   013276R         ..LL    014410R         ..SLSH  013332R         ..99    013704R
WDFR    000116R          ..COLN  013734R         ..LPAR  013202R         ..SPC   012746R

. ABS.  000000      000
        016212      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  8480 WORDS  ( 34 PAGES)
DYNAMIC MEMORY:  9786 WORDS  ( 37 PAGES)
ELAPSED TIME:  00:02:47
```